Online Advertisement Allocation in the Presence of Customer Choices

Xiaolong Li
National University of Singapore, oralxi@nus.edu.sg

Ying Rong
Shanghai Jiao Tong University, yrong@sjtu.edu.cn

Renyu Zhang
New York University Shanghai, renyu.zhang@nyu.edu

Huan Zheng
Shanghai Jiao Tong University, zhenghuan@sjtu.edu.cn

February 17, 2021

Advertisement, as a major revenue source of e-commerce platforms, is an important online marketing tool for sellers thereof. In this paper, we explore the dynamic ad allocation with limited slots upon each customer arrival for e-commerce platforms when the advertisers specify, for each ad, the budget constraint, the time periods when the ad should be displayed, and the click-through (lower-limit) constraint for certain customer segments. The goal of the platform is to maximize its payoff over the entire horizon. We propose a two-stage stochastic program framework in which the platform first decides the click-through goals for each ad/customer-type pair, and then devises the ad allocation policy to satisfy these goals in the second stage. We show that the optimal click-through goals can be achieved efficiently by solving a convex program, which can further reduce to a scalable linear program if the customer click-through behavior follows the multinomial logit model. Moreover, we provide a family of debt-weighted algorithms to achieve the optimal click-through goals, and prove that they are asymptotically optimal when the problem size scales to infinity. Compared to choice-based linear programming and its variant, our approach has better scalability and can deplete the ad budgets more smoothly throughout the horizon, which is very much desirable for the online advertising business in practice.

Key words: Online Advertising Platform, Choice Models, Online Convex Optimization

1. Introduction

The recent 10 years have witnessed a rapid growth of internet technology and smartphone penetration, which have driven online advertising to become an unprecedentedly enormous industry with a substantial impact on the entire economy. The Interactive Advertising Bureau[1] reports that online advertising revenue in the United States market has increased to \$124.6 billion in 2019 (a 16% year-over-year growth rate, and a 19% average annual growth rate since 2010). Indeed, the primary

[1] See `https://www.iab.com/insights/internet-advertising-revenue-fy2019-q12020/` for more details.

approach taken by large online platforms such as Facebook to monetize their large-scale user traffic is online advertising. For example, in 2019, Facebook earned $69.7 billion revenue from advertising, which consists of 98.53% of its total revenue[2].

Among others, an important format of online advertising is the e-commerce advertising. E-commerce advertising refers to the form of online marketing or advertising that drives top-of-tunnel traffic to convert into product sales. For instance, Amazon Advertising provides "sponsored products"[3], with which advertisers pay Amazon to promote their products by listing the ads both within the shopping results and on the product pages (see, Figure 1). The Sponsored Product ads use the cost-per-click (CPC) mechanism under which advertisers pay an advertising fee to the platform when customers click their ads. Advertisers choose the campaign budgets and how much to bid per click for their ads. E-commerce platforms, such as Alibaba, often ask advertisers to submit their advertising plans on a daily basis, specifying the time intervals (in hours) of a day during which the ads are displayed, and the corresponding cost per click. We refer interested readers to Appendix E for a detailed description of the e-commerce advertising business on Alibaba. Amazon also allows advertisers to set the keywords and products so that the ad can be more efficiently matched with customer queries. Alternatively, advertisers can select automatic targeting to allow Amazon to match their ads to relevant search terms and products. Indeed, the advertising service is an important source of revenue for Amazon, primarily included in service offerings that contributed 14.1 billion US dollars (i.e., 5.02%) to its *annual net sales* in 2019[4]. As another example, Facebook launched the service "Dynamic Ads" to promote the products of advertisers to people who expressed interest on relevant keywords or similar products[5]. The Dynamic Ads will automatically choose products from the catalog provided by the advertisers and display them to customers.

A large-scale e-commerce advertising platform (e.g., Alibaba, Amazon or Facebook Marketplace) generally runs thousands of advertising campaigns for different advertisers simultaneously. Each campaign is usually associated with a time interval when the ad should be displayed to customers, a budget that the advertiser wishes to spend as much as possible during the campaign horizon, and a bid for cost-per-click that dictates how much budget of the ad should be deducted upon each user click. The advertising platform dynamically allocates its ad spaces (i.e., customer impressions) to the ads whose campaigns are active in order to maximize the total advertising value subject to the budget constraints of the ads. As discussed above, an advertiser may require the platform to target

[2] See the 2019 financial report of Facebook: `https://www.sec.gov/ix?doc=/Archives/edgar/data/1326801/000132680120000013/fb-12312019x10k.htm`.

[3] `https://advertising.amazon.com/solutions/products/sponsored-products`

[4] See the 2019 financial report of Amazon: `https://www.sec.gov/ix?doc=/Archives/edgar/data/1018724/000101872420000004/amzn-20191231x10k.htm`

[5] See `https://www.facebook.com/business/help/397103717129942?id=1913105122334058` for more details

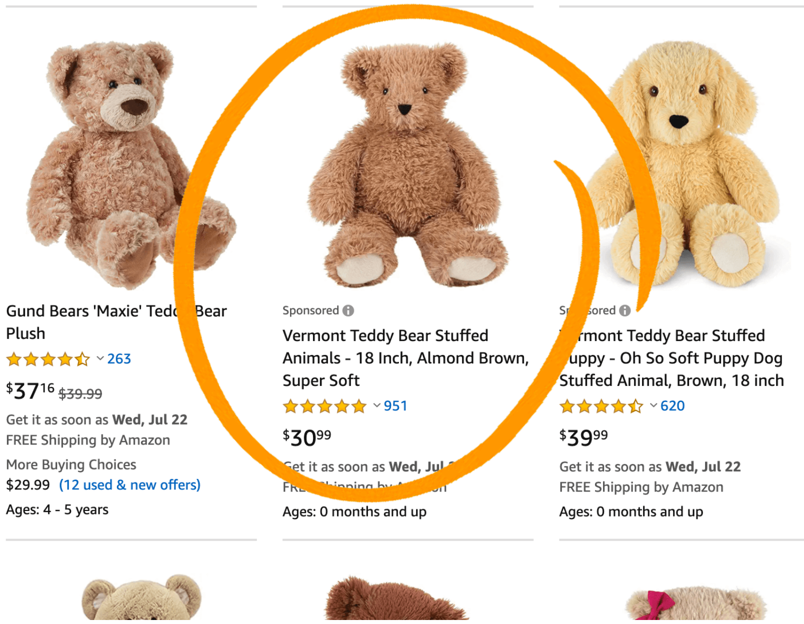


Figure 1 An Example of Sponsored Products on Amazon

his/her advertising campaign and ads to specific customer segments (location, age, gender, social status, etc.). It is also not uncommon for advertisers and, thus, the platform to set click-through constraints for the ads (i.e., the minimal number of click-throughs during the entire campaign of an ad). For instance, Microsoft provides a Partner Incentive Cooperative Marketing Fund to subsidize its partners in whose website the number of click-throughs for Microsoft's ad is above 250 during the promotion events (Microsoft 2020). In addition, from a long-term perspective, the number of click-throughs for an ad has substantial impact on the long-term retention of the advertiser, which prompts the platform to devise the ad allocation policy to secure a certain number of click-throughs for each ad (see, e.g., Ye et al. 2020).

To efficiently allocate its ad spaces, an online advertising platform faces a central operations problem to dynamically select a set of ads, which we refer to as an offer set, displayed to each arriving customer in order to generate the highest total value throughout the planning horizon. On one hand, under the customer choices when an offer set is displayed, the platform has to carefully balance the notorious trade-off in assortment optimization between expanding the offer set to enlarge the market share and keeping it small to reduce the cannibalization between different ads thereof. On the other hand, the differentiated active periods of the ads and the click-through constraints further require the platform to wisely allocate impressions from different customer segments to the ads.

The main goal of this paper is to explore the e-commerce ad allocation of an advertising platform. The key question we seek to address is: How should the platform dynamically personalize the ad offer sets of each customer impression to maximize the total value from advertising throughout a planning horizon in the presence of differentiated active time periods, advertising budgets, and click-through constraints of the ads? We present a general stochastic program model to study this complex dynamic ad allocation problem and propose a family of scalable algorithms to identify the optimal ad allocation policy. More specifically, we relax the problem as a two-stage stochastic program in which the platform first decides the click-through goals for each ad/customer-type pair, and then devise the ad offer set policy to satisfy these goals in the second stage. We show that simple debt-weighted offer-set policies could satisfy the (endogenous) click-through goals and achieve the optimal (asymptotic) value. We also conduct extensive numerical experiments to uncover several interesting insights about e-commerce ad allocation and the benefits of our proposed algorithms thereof.

1.1. Main Contributions

The contributions of this paper can be summarized as follows:

Two-Stage Stochastic Program Framework. We develop in this paper a new two-stage stochastic program modeling framework to study online ad allocation for an e-commerce platform. Specifically, we consider the platform deciding the click-through goals for each ad/customer-type in the first stage and the offer set in the second stage. This novel framework generalizes the classic assortment optimization with limited inventory literature (e.g., Liu and Van Ryzin 2008) to incorporating heterogeneous active time periods of the ads. Our framework also enables us to characterize the necessary and sufficient condition under which the click-through goals are feasible. The characterization of feasible click-through goals helps uncover interesting insights on the operational and managerial implications of click-through constraints on ad allocation that, to meet the click-through goals, the platform should also account for the non-click possibilities of the customers.

Debt-Weighted Offer Set Algorithms. As our main contribution, we propose a family of simple and effective algorithms, referred to as *debt-weighted offer set policies*, and demonstrate their optimality. The policies assign a "debt" to each click-through goal, which measures the difference between the realized total click-throughs and the endogenous goals set in the first period. Then, an offer set optimization problem is solved that maximizes a debt-weighted value function upon the arrival of each customer type. The "debt" can be either calculated dynamically or randomly sampled from a sufficiently large set constructed offline. By re-weighting the ad values with the debts associated with the click-through goals, the algorithms generate an offer set policy that satisfy the (feasible) click-through goals set in the first stage of our two-stage program.

In addition to producing offer set policies feasible for the click-through goals, our debt-weighted offer set algorithms are also asymptotically optimal as the problem size (the platform traffic and ad budget) scales up to infinity. In other words, our proposed algorithm achieves the optimal ad allocation. Through numerical experiments, we show that our algorithms perform better, more robustly, and more efficiently than some existing benchmarks in the literature for most problem instances. Our numerical experiments also demonstrate that the proposed debt-based algorithms provide much smoother depletion of budgets over the entire planning horizon than the benchmarks. This highlights the practical applicability of our approach because smooth budget depletion is a very desirable property for the real-world online advertising business.

In summary, the key takeaway from this paper is that the proposed two-stage stochastic program modeling framework together with the associated debt-weighted off set algorithms can efficiently address the ad allocation problem for e-commerce platforms. Our approach is simple, efficient and scalable, with a provable optimality guarantee and strong numerical performances. The rest of this paper is organized as follows. We review related literature in Section 2. The general modeling framework is introduced in Section 3 and the necessary and sufficient condition for the second-stage offer set policy is proposed in Section 4. We study the optimal ad allocation policy in Section 5 and present the numerical studies in Section 6. Section 7 concludes this paper. All proofs are relegated to the Online Appendices.

2. Literature Review

This paper proposes a general modeling framework and efficient algorithms to study optimal online ad allocation for an e-commerce platform. We are primarily related to three streams of research in the literature: (a) ad allocation for online advertising platform, (b) resource allocation with individualized service level constraints and (c) (dynamic) personalized assortment optimization. Papers in the literature generally focus on one perspective of the three topics above, whereas our work makes a contribution to all three streams of literature jointly.

Ad allocation is a notoriously challenging problem for an online advertising platform. Scheduling advertisement display on websites has been widely studied in the literature. For example, Adler et al. (2002) design algorithms to help a publisher determine the best subset of ads to share the limited space on web pages, Chickering and Heckerman (2003) identify an ad display schedule to maximize the expected number of clicks with inventory constraints, whereas Kumar et al. (2006) seek to maximize the revenue given a planning horizon. Feldman et al. (2009) study an online ad allocation problem with a set of advertisers and online arrival ad impressions, the matches between which have been weighted based on contracts, and propose a policy to maximize the total weight of the assigned edges. Walsh et al. (2010) propose an ad inventory allocation algorithm for the scenario that the number of segments grows exponentially in the number of customer features. Yang et al. (2012) combine an ad inventory allocation problem with multi-objective of revenue and fairness. For maximizing reach of customers and minimizing variance of the outcome simultaneously in targeted advertising, Turner (2012) formulates an planning problem with a quadratic objective to spread ads across all targeted customer types. Balseiro et al. (2014) formalize an ad exchange problem as a multi-objective stochastic control problem considering both the revenue from exchange and click through rates, and derive an efficient policy for online ad allocation with uncertainty. Also for dealing with uncertainty, Shen et al. (2020) propose an integrated planning model with distributionally robust chance-constrained program in online ad allocation. We refer interested readers to Choi et al. (2020) for a comprehensive review of this literature. The main modeling difference of our paper from this literature is that, using choice models, we clearly model the click-through behaviors of a customer in the presence of multiple ads displayed to him/her simultaneously. Furthermore, we propose a new two-stage stochastic program framework to study the ad allocation problem, and design a family of debt-weighted ad offer set policies that generate the optimal value for the platform.

The resource allocation problem to meet service target constraints in the face of uncertain demand has been extensively studied in the inventory literature (see, e.g., Eppen 1979, Swaminathan and Srinivasan 1999, Alptekinoğlu et al. 2013). Substantial recent progress has been made to contend with this problem using the approaches inspired by online convex optimization (Hazan 2019) and Blackwell's approachability theorem (Blackwell 1956). For example, Hou et al. (2009) study the single-resource allocation in wireless networks with quality of service (QoS) constraints, which are essentially the same as the type-II service-level constraints in the inventory literature. In a similar vein, Zhong et al. (2017) characterize the optimal safety-stock level with individual type-II service-level constraints. Lyu et al. (2019) and Lyu et al. (2017), respectively, extend both the approach and results to the context of type-I service-level constraint and process flexibility. Utilizing a semi-infinite linear program formulation, Jiang et al. (2019) generalize and unify models in this literature and propose a simple randomized rationing policy to meet general service-level constraints, including type-I and type-II constraints, and beyond. Ma and Xu (2020) consider an online matching problem with concerns of agent-group fairness, define two different service-level objectives, instead of maximizing the number of matches, as the metrics of long-run and short-run fairness, and show competitive ratios of their algorithms. Our contribution to this literature is that we generalize the concept of service-level constraints to incorporate customer choice uncertainty and indirect resource allocation through assortment planning. We also propose a family of debt-weighted offer set algorithms and demonstrate their optimality to meet the service-level constraints and to generate the total payoff for the platform.

In the recent decade, online e-commerce platforms have typically provided numerous products for customers to choose from (Feldman et al. 2018). Manufacturing firms have also expanded their product lines due to business trends (e.g., fast fashion, Caro et al. 2014) or technology revolution (e.g., 3D printing, Dong et al. 2017). The ever-expanding product pool makes personalized assortment more attractive. Therefore, personalized assortment optimization has also received growing attention in the literature. Leveraging the competitive ratio framework, Golrezaei et al. (2014) propose inventory-balancing algorithms that guarantee the worst-case revenue performance without any forecast of the customer type distribution. Bernstein et al. (2019) combine dynamic assortment planning, demand learning, and customer type clustering in a Bayesian framework and propose a prescriptive assortment personalization approach for online retailing. Using re-solving heuristics, Jasin and Kumar (2012) study dynamic personalized assortment planning in a network revenue management framework and show that the proposed heuristics achieve a constant optimality loss. Kallus and Udell (2020) consider a dynamic assortment personalization problem with a large number of items and customer types as a discrete-contextual bandit problem and propose a structural approach with efficient optimization algorithms. Chen et al. (2016) formalize a new checkout recommender system at Walmart's online grocery as an online assortment optimization problem with limited inventory and propose an inventory-protection algorithm with a bounded competitive ratio. A general personalized resource allocation model with customer choices is studied by Gallego et al. (2016). Adopting the column-generation approach to solve the choice based linear program, the authors introduce algorithms with theoretical performance guarantees. Considering the uncertainty in estimating the MNL choice model, Cheung and Simchi-Levi (2017) propose a Thompson Sampling based policy to estimate the latent parameters by offering personalized assortment and demonstrate its near optimality. Our contribution towards this literature is that we propose a new two-stage stochastic program framework to study the ad offer set optimization problem. Moreover, we design a family of debt-weighted offer set policies that prove to be asymptotically optimal and generate revenue (value) with lower variance than the benchmarks in the literature such as choice-based linear program (CBLP) and the linear programming with re-solving (LP re-solving) heuristics.

3. Model

We consider an e-commerce platform such as Alibaba, Amazon or Facebook Marketplace, which matches its user traffic with both organic product recommendations and advertisements. The advertisements thereof are usually labeled as *Sponsored Products*, as shown in Figure 1. The planning horizon is a single day and consists of Σ periods denoted as $\{1,2,...,\Sigma\}$, where one period can be interpreted as the shortest time horizon in which advertisers can plan their campaigns. There are T customers arriving at the platform in a day, $T_s = \zeta_s T$ of which arrive in period s, where $\sum_{s=1}^{\Sigma} \zeta_s = 1$. At the beginning of the day, advertisers launch a set of ad campaigns, which we denote as $\mathcal{N} := \{1,2,...,n\}$. For each ad $i \in \mathcal{N}$, its advertiser also set the time periods in which the campaign is active on the platform. We use $\mathcal{N}_s$ to denote the set of active ad campaigns in period s, and $n_s := |\mathcal{N}_s|$ to denote the number of active ads in period s. The organic recommendation (i.e., the recommended items not labeled as "Sponsored" in Figure 1) is usually conducted by the recommendation department, a different team NOT responsible for advertising strategies. Furthermore, the ads are placed in some specific slots exclusively allocated to advertising. In this paper, we focus on the ad assignment problem and, therefore, treat the organic recommendations as exogenous. More specifically, we use ad 0 to denote the representative organic recommendation, which is always displayed to each customer upon arriving at the platform. Hence, ad 0 may represent multiple products offered to the customer. We define $\bar{\mathcal{N}} = \mathcal{N} \cup \{0\}$ and $\bar{\mathcal{N}}_s = \mathcal{N}_s \cup \{0\}$ for each s. Associated with each ad campaign $i \in \mathcal{N}$ is a vector of parameters $\lambda_i = (\sigma_i, K_i, B_i, b_i)$, where σ_i is the starting period, K_i is the length (in periods), $B_i > 0$ is the total budget, and $b_i > 0$ is the bid price set by the advertiser for this ad campaign. More specifically, campaign i is activated on the platform at the beginning of period σ_i and remains active thereof until the start of period $\sigma_i + K_i$. B_i is the maximum amount of advertising fee the advertiser will pay the platform throughout the ad campaign's life time, and the budget will be depleted by b_i upon each click by a customer of the platform. Hence, the platform adopts the cost-per-click mechanism, which is commonly used in online advertising. For ad 0 (i.e., the organic recommendation), the advertiser does not pay the platform to be displayed, so we set $b_0 = 0$ and $B_0 = +\infty$. Without loss of generality, we also set $\sigma_0 = 1$ and $K_0 = \Sigma$. The campaigns' information need to be determined by advertisers day ahead as shown in the left part of Figure 2 (See Appendix E for the detail about how to set up campaigns in Alibaba).

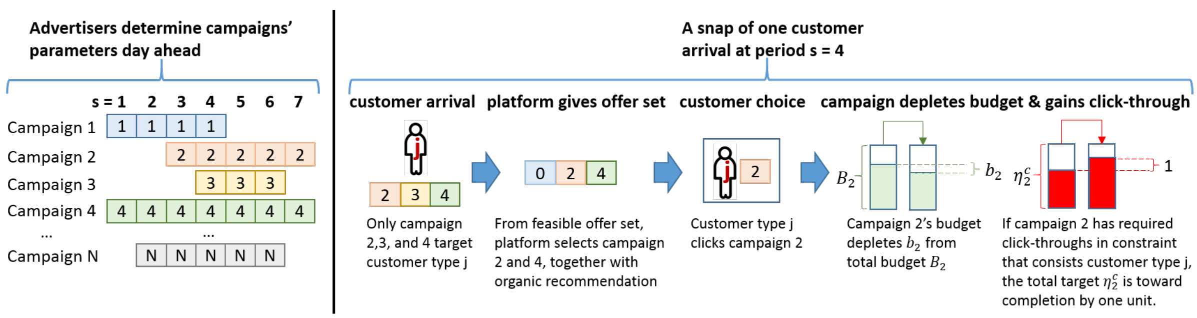


Figure 2 System Illustration

In each period s, T_s customers (also called user views, UVs) arrive at the platform in a sequential manner over the horizon $\mathcal{T}_s = \{1, 2, ..., T_s\}$. In period s, for each customer t, her type ξ_s^t is *i.i.d.* and follows a discrete distribution on $\mathcal{M}$, with $\mathbb{P}(\xi_s^t = j) = p_j^s$ where $j \in \mathcal{M}$ and $\sum_{j \in \mathcal{M}} p_j^s = 1$. In period s, upon the arrival of customer t, the platform decides the (possibly randomized) set of ads/sponsored products displayed to this UV, denoted by $S_s^t \subset \mathcal{N}_s$. We call the set of displayed ads/sponsored products as an *offer set*.

For a customer of type $j \in \mathcal{M}$ arriving at the platform in period s, if an offer set of ads $S \subset \mathcal{N}_s$ is displayed (see, Figure 1), she may or may not click some ads in the set S. Since the organic recommendation is always included in the offer set, we have $0 \in S$. For each customer type j, each offer set S, and each ad $i \in S$, define $y_i^j(S) \in \mathbb{Z}^+$ as the random variable referring to the total number of click-throughs a type-j customer would have on ad i when the offer set S is displayed. We denote the expected click-throughs of ad i in an offer set S from a type j customer as $\phi_i^j(S) := \mathbb{E}[y_i^j(S)]$. For now, we do not specify any structure of the customer click-through behavior except that the expected click-throughs are finite, i.e., $\phi_i^j(S) < +\infty$ for any $i \in S$ and $j \in \mathcal{M}$. Hence, our model is quite general and allows for multiple clicks of one or more ads from a customer. If $i \notin S$, by convention, $y_i^j(S) = 0$ and, thus, $\phi_i^j(S) = 0$. The right part of Figure 2 provides illustration the sequence of events after a customer of type j arrives.

From the advertisers' perspective, they target their ads to the relevant customer segments based on their past behavioral patterns, potential interests and demographic information (see, e.g., Choi et al. 2020). Specifically, each ad campaign $i \in \mathcal{N}$ is characterized by a set of customer segments $\mathcal{L}_i \subseteq \mathcal{M}$ targeted by this ad. That is, $i \notin S$ displayed to a type-j customer, if $j \notin \mathcal{L}_i$. For the organic recommendation, the platform generally does not impose any constraints on the customer segments it may be displayed to, i.e., $\mathcal{L}_0 = \mathcal{M}$. Furthermore, consistent with the practice in advertising (e.g., Microsoft 2020), the advertisers would require ad i receives at least η_i^c click-throughs throughout the day for a set of customer segments $c \subseteq \mathcal{L}_i$. That is, $\sum_{s=\sigma_i}^{\sigma_i+K_i-1} \sum_{t=1}^{T_s} \sum_{j\in c} y_i^j(S_s^t) \mathbf{1}_{\{\xi_s^t = j\}} \geq \eta_i^c$ for any $i \in \mathcal{N}$ and $c \subseteq \mathcal{L}_i$. For example, the advertiser may require the platform to target its ads for diapers to new parents. In practice, c can be $\mathcal{L}_i$ itself or a singleton belonging to $\mathcal{L}_i$. If the advertiser do not specify a particular minimal click-throughs for customer segment set c, then $\eta_i^c = 0$. In this case, the constraint associated with an inactive set c can be removed. Without loss of generality, $\eta_0^c = 0$ for each set $c \subset \mathcal{M}$. In many scenarios, the advertising contract specifies that the minimal click-throughs have to be met. For instance, Microsoft (as an advertiser) requires its partners (i.e., the online advertising platforms where Microsoft runs its advertising campaigns) to earn at least 250 click-throughs during one ad campaign to be qualified to receive the support through its partner incentives cooperative marketing fund.

In period s and with a customer of type $\xi_s^t = j$ arriving, the platform displays the offer set $S_s^t \in \mathcal{S}_j \subset 2^{\mathcal{N}_s}$ to him/her, where $\mathcal{S}_j$ is the collection of all possible offer sets for a customer of type j and $2^{\mathcal{N}_s}$ is the power set of $\mathcal{N}_s$. Hence, if $\xi_s^t = j$, then for any $i \in S_s^t \in \mathcal{S}_j$, we must have $j \in \mathcal{L}_i$, i.e., a customer of any type must be in the target set for any ad displayed to her. Furthermore, the organic recommendation is always included in the offer set, i.e., $0 \in S_s^t$. As introduced above, the advertiser of ad i pays the platform b_i for each click it receives and the total payment of this ad in day s and period t is, therefore, $b_i y_i^{\xi_s^t}(S_s^t)$. The objective of the platform is to maximize its total payoff over the day by matching the n ads with T customers. We denote the value of each click of ad $i \in \bar{\mathcal{N}}$ as r_i. The interpretation of r_i can be quite general and include the following scenarios as special cases. For the case where the platform seeks to maximize its total advertising revenue, $r_i = b_i$ for each $i \in \mathcal{N}$. For the case where the platform seeks to maximize the total value of its clients (i.e., advertisers, see, e.g., Hao et al. 2020), r_i is interpreted as the value of one click-through for ad i to its advertiser. In particular, r_0 is the value per click for the organic recommendation. For example, r_0 can be the average commission fee the seller pays for one click-through of the organic recommendation. For an e-commerce platform, r_0 is in general one order of magnitude lower than r_i ($i \in \mathcal{N}$). We also remark that all our results can be easily generalized to the case where the value is customer-segment-specific as well, i.e., the value of a click from a type-j customer for ad i is denoted by r_{ij}.

We are now ready to formulate the platform's ad assignment problem as a multi-period stochastic program. Specifically, the platform seeks to optimize the total expected value throughout the day:

$$\begin{aligned}
&\max \mathbb{E}\left[\sum_{s=1}^{\Sigma}\sum_{t=1}^{T_s}\sum_{i\in\bar{\mathcal{N}}_s} r_i y_i^{\xi_s^t}(S_s^t)\right] \\
&\text{s.t.} \sum_{s=\sigma_i}^{\sigma_i+K_i-1}\sum_{t=1}^{T_s} b_i y_i^{\xi_s^t}(S_s^t) \le B_i, \ \textit{almost surely} \text{ for each } i\in\mathcal{N}, \\
&\qquad \sum_{s=\sigma_i}^{\sigma_i+K_i-1}\sum_{t=1}^{T_s}\sum_{j\in c} y_i^j(S_s^t)\mathbf{1}_{\{\xi_s^t=j\}} \ge \eta_i^c, \ \textit{almost surely} \text{ for each } i\in\mathcal{N} \text{ and } c\subseteq\mathcal{L}_i, \\
&\qquad 0\in S_s^t \subseteq \bar{\mathcal{N}}_s \text{ for each } s,t.
\end{aligned} \tag{1}$$

where the first constraint refers to the budget constraint of each ad and the second refers to the click-through requirement for each ad with respect to different sets of customer segments, c. This is a complex stochastic programming problem. Note that (1) is equivalent to the following scaled version that maximizes the per-customer value.

$$\begin{aligned}
&\max \mathbb{E}\left[\frac{1}{T}\sum_{s=1}^{\Sigma}\sum_{t=1}^{T_s}\sum_{i\in\bar{\mathcal{N}}_s} r_i y_i^{\xi_s^t}(S_s^t)\right] \\
&\text{s.t.} \frac{1}{T(i)}\sum_{s=\sigma_i}^{\sigma_i+K_i-1}\sum_{t=1}^{T_s} b_i y_i^{\xi_s^t}(S_s^t) \le \frac{B_i}{T(i)}, \ \textit{almost surely} \text{ for each } i\in\mathcal{N}, \\
&\qquad \frac{1}{T(i)}\sum_{s=\sigma_i}^{\sigma_i+K_i-1}\sum_{t=1}^{T_s}\sum_{j\in c} y_i^j(S_s^t)\mathbf{1}_{\{\xi_s^t=j\}} \ge \frac{\eta_i^c}{T(i)}, \ \textit{almost surely} \text{ for each } i\in\mathcal{N} \text{ and } c\subseteq\mathcal{L}_i,
\end{aligned} \tag{2}$$

where $T(i) := \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s$ is the total number of potential customers for ad campaign $i\in\mathcal{N}$.

To simplify the ad allocation problem (2), we introduce an auxiliary decision vector $\boldsymbol{\alpha} = (\alpha_i^j(s) \ge 0 : i\in\bar{\mathcal{N}}, j\in\mathcal{M}, 1\le s\le\Sigma)$, where $\alpha_i^j(s)$ refers to the per-user number of click-throughs for ad i from type-j customers in period s. Hence, $\alpha_i^j(s) > 0$ only if $j \in \mathcal{L}_i$ and $\sigma_i \le s \le \sigma_i + K_i - 1$. Then, we can reformulate (2) as follows:

$$
\begin{aligned}
\max_{\boldsymbol{\alpha}} \ & \sum_{s=1}^{\Sigma} \zeta_s \sum_{i \in \bar{\mathcal{N}}_s} r_i \sum_{j \in \mathcal{L}_i} \alpha_i^j(s) \\
\text{s.t. } & \frac{1}{T_s} \sum_{t=1}^{T_s} y_i^j(S_s^t) \mathbf{1}_{\{\xi_s^t = j\}} \ge \alpha_i^j(s), \ \textit{almost surely} \text{ for each } i \in \bar{\mathcal{N}}, j \in \mathcal{M}, s \\
& \sum_{s=\sigma_i}^{\sigma_i + K_i - 1} \zeta_s \sum_{j \in \mathcal{L}_i} \alpha_i^j(s) \le \frac{B_i}{T b_i}, \text{ for each } i \in \mathcal{N}, \\
& \sum_{s=\sigma_i}^{\sigma_i + K_i - 1} \zeta_s \sum_{j \in c} \alpha_i^j(s) \ge \frac{\eta_i^c}{T}, \quad \text{for each } i \in \bar{\mathcal{N}}, c \subseteq \mathcal{L}_i,
\end{aligned}
\tag{3}
$$

where $\mathbf{1}_{\{\cdot\}}$ is the indicator function. With formulation (3), we can interpret $\alpha_i^j(s)$ as the (virtual) goal of per-user click-throughs for ad i from type-j customers. Next, we relax (3) to a two-stage stochastic program by replacing the first (sample-path) constraint in (3) to a constraint on the expected click-throughs. Specifically, in the first stage, the platform decides the click-through goals $\boldsymbol{\alpha}$, and, in the second stage, it selects the offer set S_s^t displayed to each customer t in period s throughout the horizon. Define $\tilde{G}$ as a randomized offer-set policy (in the second stage) that maps customer type ξ_s^t to a random offer set $S_s^t = S_s(\xi_s^t | \tilde{G}) \in \mathcal{S}_{\xi_s^t}$. The set of all feasible (randomized) offer-set policies is denoted by $\tilde{\mathcal{G}}$. We denote the offer-set policy $\tilde{G}$ constrained in period s as $\tilde{G}_s$ and, accordingly, the set of all randomized policies in period s as $\tilde{\mathcal{G}}_s$. With $\tilde{G} = (\tilde{G}_1, ..., \tilde{G}_\Sigma)$ and $S_s(\xi_s^t | \tilde{G}_s) := S_s(\xi_s^t | \tilde{G})$, the stochastic program (3) can be relaxed into the following formulation,

$$
\begin{aligned}
\max_{\tilde{G}, \boldsymbol{\alpha}} \ & \sum_{s=1}^{\Sigma} \zeta_s \sum_{i \in \bar{\mathcal{N}}_s} r_i \sum_{j \in \mathcal{L}_i} \alpha_i^j(s) \\
\text{s.t. } & \mathbb{E}[y_i^j(S_s(\xi_s | \tilde{G}_s)) \mathbf{1}_{\{\xi_s = j\}}] \ge \alpha_i^j(s), \ \textit{almost surely} \text{ for each } i \in \bar{\mathcal{N}}, j \in \mathcal{M}, s \\
& \sum_{s=\sigma_i}^{\sigma_i + K_i - 1} \zeta_s \sum_{j \in \mathcal{L}_i} \alpha_i^j(s) \le \frac{B_i}{T b_i}, \text{ for each } i \in \mathcal{N}, \\
& \sum_{s=\sigma_i}^{\sigma_i + K_i - 1} \zeta_s \sum_{j \in c} \alpha_i^j(s) \ge \frac{\eta_i^c}{T}, \quad \text{for each } i \in \bar{\mathcal{N}}, c \subseteq \mathcal{L}_i, \\
& \tilde{G}_s \in \tilde{\mathcal{G}}_s, \quad \text{for each } s,
\end{aligned}
\tag{4}
$$

where ξ_s follows the same distribution as ξ_s^t, and the expectation is taken with respect to the joint distribution of customer type ξ_s, offer-set policy $\tilde{G}$, and customer click-throughs $y_i^j(S)$.

Let us now take a closer look at the two-stage stochastic program (4). In the first-stage, knowing the distribution of customer type $\mathcal{P}$ and the click-through expectations $\Phi := \{\phi_i^j(S) : j \in \mathcal{M}, i \in S \subset \bar{\mathcal{N}}\}$, but not the realization of customer type ξ_s and click-throughs $y_i^j(S)$, the firm decides the virtual goals of the per-period click-through expectations $\boldsymbol{\alpha} = \{\alpha_i^j(s) : i \in \bar{\mathcal{N}}, j \in \mathcal{M}, 1 \le s \le \Sigma\}$.

In the second-stage, the customer type in period s, ξ_s, realizes, after which the firm decides the offer set displayed to the customer according to a general (randomized) policy $\tilde{G} \in \tilde{\mathcal{G}}$. We denote $S_s^j(\tilde{G}, \xi_s) = S_s(\xi_s|\tilde{G})\mathbf{1}_{\{\xi_s = j\}}$ as the (randomized) offer set, under policy $\tilde{G}$, displayed to a type-j customer if the realized customer type is ξ_s. Hence, if the realized customer type $\xi_s \neq j$, $S_s^j(\tilde{G}, \xi_s) = \emptyset$. The first constraint of (4) dictates that the platform is obliged to meet the click-through *goals* in each period s, i.e., under the offer set policy $\tilde{G}$,

$$\mathbb{E}[y_i^j(S_s^j(\tilde{G}, \xi_s))] \geq \alpha_i^j(s), \text{ for each } i \in \bar{\mathcal{N}}, j \in \mathcal{M}, 1 \leq s \leq \Sigma. \tag{5}$$

In period s, note that $S_s^j(\tilde{G}, \xi_s)$ is the offer set prescribed by policy $\tilde{G}$ displayed to a type-j customer when the realized type is ξ_s, which can be equivalently represented by a random vector $x^j(\tilde{G}, \xi_s, s) := (x_0^j(\tilde{G}, \xi_s, s), x_1^j(\tilde{G}, \xi_s, s), x_2^j(\tilde{G}, \xi_s, s), ..., x_n^j(\tilde{G}, \xi_s, s))' \in \{0,1\}^{n+1}$, where $x_i^j(\tilde{G}, \xi_s, s) = 1$ if and only if $j = \xi_s$ and $i \in S_s^j(\tilde{G}, \xi_s)$. The random vector representation facilitates us to impose some natural constraints for the offer set decision S_s^t. For example, the cardinality constraint of the offer set (Rusmevichientong et al. 2010, Wang 2012) can be formulated as: $\sum_{i \in \bar{\mathcal{N}}} x_i^j(\tilde{G}, \xi_s, s) \leq K$ for all $j \in \mathcal{M}$, where K is the maximum size of an offer set displayed to any customer. More generally, the totally unimodular constraint (Davis et al. 2013), which includes the cardinality constraint as a special case, can be incorporated as: $A \cdot x^j(\tilde{G}, \xi_s, s) \leq b$ for all $j, \xi_s \in \mathcal{M}$ and s, where A is a totally unimodular matrix.

To conclude this section, we reformulate the platform's problem (4) back to a periodic-review infinite horizon problem as (6), which will prove useful in the characterization of the optimal dynamic ad allocation policy.

$$\begin{aligned} \max_{\tilde{G}, \boldsymbol{\alpha}} \ & \sum_{s=1}^{\Sigma} \zeta_s \sum_{i \in \bar{\mathcal{N}}_s} r_i \sum_{j \in \mathcal{L}_i} \alpha_i^j(s) \\ \text{s.t. } & \liminf_{T_s \uparrow +\infty} \frac{1}{T_s} \sum_{t=1}^{T_s} y_i^j(S_s(\xi_s^t|\tilde{G}))\mathbf{1}_{\{\xi_s^t = j\}} \geq \alpha_i^j(s), \ \textit{almost surely} \text{ for each } i \in \bar{\mathcal{N}}, j \in \mathcal{M}, s \\ & \sum_{s=\sigma_i}^{\sigma_i + K_i - 1} \zeta_s \sum_{j \in \mathcal{L}_i} \alpha_i^j(s) \leq \frac{B_i}{Tb_i}, \text{ for each } i \in \mathcal{N}, \\ & \sum_{s=\sigma_i}^{\sigma_i + K_i - 1} \zeta_s \sum_{j \in c} \alpha_i^j(s) \geq \frac{\eta_i^c}{T}, \quad \text{for each } i \in \bar{\mathcal{N}}, c \subseteq \mathcal{L}_i, \\ & \tilde{G} \in \tilde{\mathcal{G}}. \end{aligned} \tag{6}$$

Note that similar periodic-review reformulation has also been adopted in the resource allocation and inventory pooling literature (e.g. Zhong et al. 2017, Jiang et al. 2019).

4. Second-Stage Analysis: Necessary and Sufficient Conditions

To solve the stochastic program (4), we start with a complete characterization of the necessary and sufficient condition for devising an offer set policy $\tilde{G}$ in the *second-stage* of the stochastic program to satisfy the click-through goals (5). This necessary and sufficient condition can reduce the search of optimal click-through goals to a convex optimization problem. Specifically, if the customer's click behavior follows the Multi-Nomial Logit (MNL) choice model, the necessary and sufficient condition can be characterized as a solution to a simple linear program.

4.1. Necessary and Sufficient Feasibility Condition in the Second Stage

We observe that the click-through goal constraints (5) can be decomposed into one in each period s. Hence, we focus on the characterizing the feasibility condition for the click-through goals in each period s separately. To characterize the feasibility of the click-through goals $\boldsymbol{\alpha}(s)$ in the second stage of the stochastic program (i.e., under what condition of the click-through goals $\boldsymbol{\alpha}(s) = (\alpha_i^j(s) : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$, such that there exists an offer set policy $\tilde{G}_s$ to satisfy all goals), we consider the following formulation with a constant objective function for each period s:

$$\begin{aligned} &\max_{\tilde{G}_s} 0 \\ &\text{s.t. } \mathbb{E}[y_i^j(S_s^j(\tilde{G}_s, \xi_s))] \geq \alpha_i^j(s), \text{ for each } i \in \bar{\mathcal{N}} \text{ and } j \in \mathcal{M} \\ &\qquad \tilde{G}_s \in \tilde{\mathcal{G}}_s \end{aligned} \tag{7}$$

We now seek to understand when the stochastic program (7) has a feasible solution. The formulation (7) is non-linear, so we first reformulate it as a linear program (LP). Define the set of all possible deterministic policies as $\mathcal{G}$, where $G \in \mathcal{G}$ is a plausible deterministic policy. Accordingly, the set of all deterministic policies in period s, G_s, is denoted by $\mathcal{G}_s$. Note that $\mathcal{G}_s$ is a finite set whose cardinality is given by $|\mathcal{G}_s| = \prod_{j\in\mathcal{M}} |\mathcal{S}_j|$, where $|\mathcal{S}_j|$ is the cardinality of $\mathcal{S}_j$. Hence, $|\mathcal{G}| = \prod_s |\mathcal{G}_s| = \left(\prod_{j\in\mathcal{M}} |\mathcal{S}_j|\right)^{\Sigma}$. Hence, a randomized policy $\tilde{G}_s(\mu_s) \in \tilde{\mathcal{G}}_s$ is defined by a probability measure $\mu_s(\cdot)$ on the finite set $\mathcal{G}_s$, which is essentially a probability simplex in the space $\mathbb{R}^{|\mathcal{G}_s|}$.

In period s, let $S_s^j(G_s, \xi_s)$ be the offer set, prescribed by policy G_s, displayed to a customer of type j if a customer of type ξ_s arrives (in the case $\xi_s \neq j$, $S_s^j(G_s, \xi_s) = \emptyset$). In period s, the average number of click-throughs per user for ad i by type j customers is, thus, given by

$$p_j^s \phi_i^j(S_s^j(G_s, j)).$$

Note that p_j^s is the probability to have a type-j customer's arrival in period s. Therefore, (7) can be reformulated as the following LP.

$$\begin{aligned} &\max_{\mu_s(\cdot)} 0 \\ &\text{s.t.} \sum_{G_s\in\mathcal{G}_s} \mu_s(G_s)p_j^s\phi_i^j(S_s^j(G_s,j)) \ge \alpha_i^j(s), \text{ for each } i\in\bar{\mathcal{N}} \text{ and } j\in\mathcal{M} \\ &\quad\sum_{G_s\in\mathcal{G}_s} \mu_s(G_s) = 1 \\ &\quad\mu_s(G_s)\ge 0 \text{ for all } G_s\in\mathcal{G}_s. \end{aligned} \tag{8}$$

Taking the dual of the LP (8), we obtain that:

$$\begin{aligned} &\min_{\theta_0,\theta_i^j} \{\theta_0 - \sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}} \alpha_i^j(s)\theta_i^j\} \\ &\text{s.t.} \sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}} p_j^s\phi_i^j(S_s^j(G_s,j))\theta_i^j - \theta_0 \le 0, \text{ for all } G_s\in\mathcal{G}_s \\ &\quad\theta_i^j\ge 0 \text{ for all } i\in\bar{\mathcal{N}} \text{ and } j\in\mathcal{M}. \end{aligned} \tag{9}$$

Note that (9) is equivalent to

$$\min_{\theta_i^j\ge 0}\left\{\max_{G_s\in\mathcal{G}_s}\sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}} p_j^s\phi_i^j(S_s^j(G_s,j))\theta_i^j - \sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}}\alpha_i^j(s)\theta_i^j\right\}\ge 0 \tag{10}$$

It then follows immediately from strong duality that (7) is feasible if and only if the minimum of the left-hand side of (10) is greater than or equal to 0. Based on this observation, the following theorem establishes the necessary and sufficient condition for the (second-stage) click-through goals $\boldsymbol{\alpha}(s)$.

Theorem 1. (Necessary and Sufficient Condition) *The click-through constraint in period s for the second-stage offer-set policy,* (7)*, is feasible if and only if*

$$\max_{G_s\in\mathcal{G}_s}\sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}} p_j^s\phi_i^j(S_s^j(G_s,j))\theta_i^j \ge \sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}}\alpha_i^j(s)\theta_i^j \quad \textit{for all } \theta_i^j\ge 0\ (i\in\bar{\mathcal{N}}, j\in\mathcal{M}) \tag{11}$$

Note that the left-hand side of inequality (11) can be viewed as a personalized offer set optimization problem. Specifically, for each customer type j, we seek to provide an offer set S_s^{j*} that maximizes the total revenue from this customer type with the per-click revenue of ad i equal to $p_j^s\theta_i^j$, i.e.,

$$S_s^{j*}(\theta) = \arg\max_{S\in\mathcal{S}_j}\sum_{i\in S} p_j^s\theta_i^j\phi_i^j(S) \tag{12}$$

Given the dual variables $\theta := (\theta_i^j : i\in\bar{\mathcal{N}}, j\in\mathcal{M})$, we define

$$g_s(\theta) := \sum_{j\in\mathcal{M}}\max_{S\in S_j}\sum_{i\in S} p_j^s\theta_i^j\phi_i^j(S),$$

which is the left-hand side of (11). Hence, we obtain an equivalent necessary and sufficient condition for the feasibility of click-through goals (8):

$$\begin{aligned} &\min_{\theta_i^j \geq 0} h_s(\theta|\boldsymbol{\alpha}(s)) \geq 0, \\ &\text{where } h_s(\theta|\boldsymbol{\alpha}(s)) := g_s(\theta) - \sum_{i\in\bar{\mathcal{N}}, j\in\mathcal{M}} \alpha_i^j(s)\theta_i^j \end{aligned} \tag{13}$$

Because $h_s(\cdot|\boldsymbol{\alpha}(s))$ is the maximum of a family of linear functions, it is jointly convex in θ for any $\boldsymbol{\alpha}(s)$. Therefore, checking the feasibility of the two-stage stochastic program (4) is reduced to minimizing a convex function $h_s(\cdot|\boldsymbol{\alpha}(s))$ over the quadrant $\{\theta_i^j \geq 0 : i \in \bar{\mathcal{N}}, j \in \mathcal{M}\}$. Hence, as long as the personalized offer-set optimization problem (12) is tractable, one could numerically check the feasibility of the click-through goals $\boldsymbol{\alpha}(s)$.

With the characterization of the necessary and sufficient condition (13) for the feasibility of click-through goals $\boldsymbol{\alpha}(s)$ in the second stage, we are now ready to reformulate the two-stage stochastic program (4) as the following convex program:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \; & \sum_{s=1}^{\Sigma} \zeta_s \sum_{i\in\bar{\mathcal{N}}_s} r_i \sum_{j\in\mathcal{L}_i} \alpha_i^j(s) \\ \text{s.t. } & h_s(\theta|\boldsymbol{\alpha}(s)) \geq 0 \text{ for all } \theta_i^j \geq 0 \text{ for all } i \in \mathcal{N}, j \in \mathcal{M}, s \\ & \sum_{s=\sigma_i}^{\sigma_i+K_i-1} \zeta_s \sum_{j\in\mathcal{L}_i} \alpha_i^j(s) \leq \frac{B_i}{Tb_i}, \text{ for each } i \in \mathcal{N}, \\ & \sum_{s=\sigma_i}^{\sigma_i+K_i-1} \zeta_s \sum_{j\in c} \alpha_i^j(s) \geq \frac{\eta_i^c}{T}, \;\; \text{for each } i \in \bar{\mathcal{N}}, c \subseteq \mathcal{L}_i, \end{aligned} \tag{14}$$

where the objective is to maximize the expected per-user value $\sum_{s=1}^{\Sigma} \zeta_s \sum_{i\in\bar{\mathcal{N}}_s} r_i \sum_{j\in\mathcal{L}_i} \alpha_i^j(s)$ (equivalently, the long-run average value). We denote $\boldsymbol{\alpha}^*$ as the solution to (14). Thus, $\boldsymbol{\alpha}^*$ is the optimal first-stage click-through goals for our ad allocation problem. According to Theorem 1, $h_s(\theta|\boldsymbol{\alpha}(s))$ defined by (13) being non-negative provides a necessary and sufficient condition for the click-through goals in period s, $\boldsymbol{\alpha}(s)$, to be obtainable in the expected sense. The convex program formulation (14), therefore, characterizes the optimal click-through goals $\boldsymbol{\alpha}^*$ and the associated optimal (relaxed) per-user value in the expected sense. However, two questions remain to be addressed: (a) How should we display the offer sets upon the arrival of each customer to achieve the optimal click-through goals $\boldsymbol{\alpha}^*$; and (b) Will the offer set display strategy achieving $\boldsymbol{\alpha}^*$ suffice to obtain the true (non-relaxed) optimal value? We will address both questions in Section 5 below.

4.2. Feasibility of Click-Through Goals for the MNL Choice Model

For a general click-through behavior model, $\{\phi_i^j(\cdot) : i \in \bar{\mathcal{N}}, j \in \mathcal{M}\}$, the necessary and sufficient condition for feasible click-through goals (11), or equivalently (13), may be difficult to check. In this subsection, we discuss the MNL click-through model to obtain sharper insights on the characterization of the feasible set for the second-stage offer-set display problem.

As explained in Section 3, we could use an equivalent binary variable representation of a deterministic offer set policy in period s, $G_s \in \mathcal{G}_s$. More specifically, a deterministic policy G_s can be equivalently represented by an $m(n+1)-$dimensional binary vector $\mathbf{x}(s) = (x_i^j(s) \in \{0,1\} : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$, where $x_i^j(s) = 1$ means that ad i is included in the offer set displayed to a type j customer. Hence, $x_0^j(s) = 1$ for all s and $j \in \mathcal{M}$. Denote the set of all plausible offer set representation vectors in period s as $\mathcal{X}(s)$. With a slight abuse of notation, we denote $\phi_i^j(\mathbf{x}(s))$ as the expected click-throughs of a type j customer for ad i if the offer set displayed to this customer in period s is represented by $\mathbf{x}(s) = (x_i^j(s) \in \{0,1\} : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$. Under the MNL model, we have

$$\phi_i^j(\mathbf{x}(s)) = \frac{v_i^j x_i^j(s)}{1 + \sum_{i' \in \bar{\mathcal{N}}} v_{i'}^j x_{i'}^j(s)}, \tag{15}$$

where $v_i^j > 0$ is the attractiveness of ad i to type-j customers. Applying Theorems 1 to the MNL choice model, we have the following corollary.

Corollary 1. *If the customer click-through behavior follows the MNL model, the period-s click-through goals $\boldsymbol{\alpha}(s)$ is feasible for the second-stage offer-set policy, if and only if*

$$\max_{\mathbf{x}(s) \in \mathcal{X}(s)} \sum_{i \in \bar{\mathcal{N}}, j \in \mathcal{M}} \frac{p_j^s v_i^j \theta_i^j x_i^j(s)}{1 + \sum_{i' \in \bar{\mathcal{N}}} v_{i'}^j x_{i'}^j(s)} \geq \sum_{i \in \bar{\mathcal{N}}, j \in \mathcal{M}} \alpha_i^j(s) \theta_i^j \ \textit{for all } \theta_i^j \geq 0 \ (i \in \bar{\mathcal{N}}, j \in \mathcal{M}) \tag{16}$$

Leveraging the structural properties of the MNL model, we can relax the integer constraints on $\mathbf{x}(s)$ and apply a change of variable argument (see the proof of Proposition 1 in Appendix C for details) to give the following sharper and simpler characterization (as the solution to a linear program) for the feasibility of the click-through goals under the MNL model.

Proposition 1. *If customers follow the MNL click-through model* (15) *and the size of an offer set cannot exceed K, we have $\boldsymbol{\alpha}(s)$ is feasible if and only if there exist $(y_i^j : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$ and $(z^j : j \in \mathcal{M})$ that satisfy the following:*

$$\begin{aligned}
& p_j^s v_i^j y_i^j \geq \alpha_i^j(s), \quad \forall i, j \\
& \sum_{i=0}^n v_i^j y_i^j + z^j = 1, \quad \forall j, \\
& \sum_{i=0}^n y_i^j \leq K z^j, \quad \forall j, \\
& y_i^j \leq z^j, \quad \forall i, j, \\
& 0 \leq y_i^j \leq 1, \quad \forall i, j, \\
& y_0^j = z^j, \quad \forall j,
\end{aligned} \tag{17}$$

where $z^j := 1/(1 + \sum_{i'} v_{i'}^j x_{i'}^j(s))$ *and* $y_i^j := x_i^j(s) z^j = x_i^j(s)/(1 + \sum_{i'} v_{i'}^j x_{i'}^j(s))$.

By replacing the first constraint in formulation (14) by (17) when customers follow the MNL click-through model, we can solve the first-stage problem as a linear program. Note that the feasibility condition (17) involves auxiliary decision variables $(y_i^j : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$ and $(z^j : j \in \mathcal{M})$. One may question whether we can further streamline the characterization of the feasibility condition by removing these auxiliary decision variables. To this end, we characterize the feasible region of the click-through goals in period s, $\mathcal{A}(s) := \{\boldsymbol{\alpha}(s) : (13) \text{ holds.}\}$.

PROPOSITION 2. *If customers follow the MNL choice model* (15) *and the size of an offer set cannot exceed K, the period-s feasible region for $\boldsymbol{\alpha}(s)$ is given by:*

$$\mathcal{A}(s) := \left\{\boldsymbol{\alpha}(s) \in \mathbb{R}_+^{(n+1)m} : p_j^s \geq \sum_{i'=1}^n \alpha_{i'}^j(s) + \frac{(1+v_0^j)\alpha_i^j(s)}{v_i^j}, \forall i,j, \text{ and } p_j^s \geq \sum_{i=1}^n \alpha_i^j(s) + \frac{1+v_0^j}{K-1}\sum_{i=1}^n \frac{\alpha_i^j(s)}{v_i^j}, \forall j\right\} \tag{18}$$

The characterization of $\mathcal{A}(s)$ reveals further insights. We observe that (18) is equivalent to

$$p_j^s \geq \sum_{i=1}^n \alpha_i^j(s) + \max\left\{\frac{1+v_0^j}{K-1}\sum_{i=1}^n \frac{\alpha_i^j(s)}{v_i^j}, \max_i\left\{\frac{(1+v_0^j)\alpha_i^j(s)}{v_i^j}\right\}\right\}, \text{ for all } j.$$

Here, p_j^s is the expected (per-user) traffic of type j customers in period s. Clearly, $\sum_{i=0}^n \alpha_i^j(s)$ is the total required traffic for type j customers if a customer will click one of the ad in the offer set with probability 1. In practice, however, a customer may end up not choosing any ad from the offer set, so we need some buffer traffic for type-j customers that accounts for the non-click circumstance.

More specifically, let $\mathcal{S}_i$ denote the collection of all offer sets containing ad i. Since the offer-set policy may be random, we define $\mu_j(S)$ as the probability of displaying offer set $S \subseteq \bar{\mathcal{N}}$ to type j customers. Thus, the desired click-through goal for ad i and customer-type j customer is

$$\sum_{S\in\mathcal{S}_i} \mu_j(S) \cdot \frac{v_i^j}{1+\sum_{i'\in S} v_{i'}^j} \geq \alpha_i^j(s).$$

Thus, the non-click probability of the ads for a type-j customer when ad i ($i \in \bar{\mathcal{N}}$) is offered satisfies that

$$\alpha_i^j(o) := \sum_{S\in\mathcal{S}_i} \mu_j(S) \cdot \frac{1+v_0^j}{1+\sum_{i'\in S} v_{i'}^j} \geq \frac{(1+v_0^j)\alpha_i^j(s)}{v_i^j}$$

Therefore, to ensure the click-through goal of type j customers and ad i, the traffic of customer-type j must satisfy $p_j^s \geq \sum_{i'} \alpha_{i'}^j(s) + \alpha_i^j(o) \geq \sum_{i'} \alpha_{i'}^j(s) + \frac{(1+v_0^j)\alpha_i^j(s)}{v_i^j}$ for all $i \in \bar{\mathcal{N}}$.

The cardinality constraint for the offer set size would impose an additional bound on the non-purchase probability of type-j customers. Specifically, let $\mathcal{S} := \bigcup_{i=1}^n \mathcal{S}_i$ be the set of all offer sets displayed to a customer. Because $|S| \le K$ for any $S \in \mathcal{S}$, $|\{i \in \bar{\mathcal{N}} : S \in \mathcal{S}_i\}| \le K$ for all S. Note that ad 0 is always included in the offer set. We have, given customer-type j,

$$(K-1)\sum_{S\in\mathcal{S}}\mu_j(S)\cdot\frac{1+v_0^j}{1+\sum_{i\in S}v_i^j}\ge\sum_{i=1}^n\sum_{S\in\mathcal{S}_i}\mu_j(S)\cdot\frac{1+v_0^j}{1+\sum_{i\in S}v_i^j}\ge\sum_{i=1}^n\frac{(1+v_0^j)\alpha_i^j(s)}{v_i^j}.$$

Thus, the non-purchase probability of all ads for type-j customer satisfies that

$$\alpha_o^j(s):=\sum_{S\in\mathcal{S}}\mu_j(S)\cdot\frac{1+v_0^j}{1+\sum_{i\in S}v_i^j}\ge\frac{1+v_0^j}{K-1}\sum_{i=1}^n\frac{\alpha_i^j(s)}{v_i^j}.$$

Therefore, given the cardinality constraint of an offer set, to ensure the click-through goals of type-j customers with respect to all ads, the traffic of customer-type j must satisfy $p_j^s \ge \sum_{i=1}^n \alpha_i^j(s) + \alpha_o^j(s) \ge \sum_i \alpha_i^j(s) + \frac{1+v_0^j}{K-1}\sum_{i=1}^n \frac{\alpha_i^j(s)}{v_i^j}$. In summary, the characterization for the feasibility of $\boldsymbol{\alpha}(s)$ demonstrates that, to meet the click-through goals, we should also account for the *non-click* cases.

5. Algorithms for Advertisement Allocation Optimization

In this section, we focus on devising algorithms that address the ad allocation problem (1). More specifically, we study our click-through goal framework (6), propose adaptive offer-set policies that meet the optimal click-through goals $\boldsymbol{\alpha}^*$, and demonstrate that our proposed algorithm is asymptotically optimal as the problem size scales to infinity. In the literature, there are two standard solution approaches to characterize the optimal ad allocation policy. One is dynamic programming (DP, e.g., Talluri and Van Ryzin 2004). However, the DP approach suffers from the notorious curse of dimensionality, which is worsened by the minimal click-through requirements because the state variable space needs to account for the accumulative click-throughs of each ad with respect to different sets of customer segments, c. The other commonly adopted approach is the choice-based linear programming (CBLP, see, e.g., Liu and Van Ryzin 2008) and linear programming resolving (LP-resolving, see, e.g., Jasin and Kumar 2012) heuristics. With the cardinality constraint on offered assortments, one difficulty using CBLP or LP-resolving is that the number of variables (i.e., the probability of each assortment for all customer types) quickly explodes as the number of products increases, even when the choice model is restricted to MNL. Our proposed algorithms have better scalability than DP- and LP- based approaches. In addition, compared with CBLP or LP-resolving, our algorithms can deplete the budget of each ad more smoothly throughout the horizon, which is very desirable for the advertising business in practice. For example, Google recommends a "Standard" ad delivery method for most advertisers, especially those with a low budget, to avoid exhausting their budgets early[6]. Under the "Standard" delivery, each advertisement can reach customers evenly throughout the day.

[6] See `https://support.google.com/google-ads/answer/2404248?hl=en` for more details

5.1. Feasible Long-Run Average Offer-Set Policy

Before presenting the optimal ad allocation policy for a finite horizon problem (1), we first seek to obtain a feasible long-run average offer set policy for the click-through goals of each period s, $\boldsymbol{\alpha}(s)$, satisfying condition (5). By focusing on the infinite horizon problem, we can ignore the impact of budget constraint on the offer-set policy, which will be incorporated as we move on to study the optimal ad allocation in Section 5.2. Characterizing such a feasible long-run average policy that meets the click-through goals is interesting by itself. To this end, one could solve the primal linear program (8) and obtain a feasible randomized policy $\tilde{G}_s(\mu_s)$. This approach, though intuitive, may be computationally prohibitive because the LP (8) has $\mathcal{O}(m2^n)$ decision variables and $\mathcal{O}(mn)$ constraints. Therefore, we resort to a data-driven adaptive algorithm to generate the random dual variables $\{\theta_i^j \geq 0 : i \in \bar{\mathcal{N}}, j \in \mathcal{M}\}$ and the associated offer-set policy. Specifically, we consider the infinite-horizon version of the click-through goals of period s in which customers arrive sequentially:

$$
\begin{aligned}
&\max_{\tilde{G}_s} 0 \\
&\text{s.t. } \liminf_{T_s \to +\infty} \frac{1}{T_s} \sum_{t=1}^{T_s} y_i^j(S_s^j(\tilde{G}_s, \xi_s^t)) \mathbf{1}_{\{\xi_s^t = j\}} \geq \alpha_i^j(s), \text{ for each } i \in \bar{\mathcal{N}} \text{ and } j \in \mathcal{M} \\
&\quad \tilde{G}_s \in \tilde{\mathcal{G}}_s.
\end{aligned} \tag{19}
$$

Note that Theorem 1 shows condition (5) is equivalent to condition (11). To present a feasible algorithm for the click-through goals under (11) for period-s, we slightly abuse the notation and denote $y_i^j(t,s) = y_i^j(S_s^j(\tilde{G}_s, \xi_s^t)) \mathbf{1}_{\{\xi_s^t = j\}}$ as the number of click-throughs of ad i by a type-j customer arriving the t-th in period s. We are now ready to propose the debt-weighted offer-set (DWO) policy (Algorithm 1) in period s, denoted as $\tilde{G}_{DWO}(s)$, which is feasible if (13) holds.

Similar to checking the feasibility of the click-through goals $\boldsymbol{\alpha}(s)$, the DWO policy involves solving an offer-set optimization (20) upon the arrival of each customer. The personalized offer-set $S_s^*(t)$ can be efficiently found for a broad class of choice models such as multinomial logit (MNL) and independent choices (i.e., $y_i^j(S)$ is independently distributed for different $i \in S$). We call Algorithm 1 the *debt-weighted offer-set* policy because the offer-set optimization is weighted by the "debt" of each customer-advertisement pair for customers $\{1, 2, ..., t-1\}$. Note that $(t-1)\alpha_i^j(s)$ is the click-through goal of ad i by type-j customers until customer t in period s, whereas $\sum_{\tau=1}^{t-1} y_i^j(\tau, s)$ is the total realized click-throughs by then. Therefore, $\left(d_i^j(t,s)\right)^+ = \max\left((t-1)\alpha_i^j(s) - \sum_{\tau=1}^{t-1} y_i^j(\tau,s), 0\right)$ is the total "debt" owed by the platform to the desired click-through goal associated with ad i and customer-type j for customer t in period s. In addition, note that the debts of customer t in period s, $\{d_i^j(t,s) : i \in \mathcal{N}, j \in \mathcal{M}\}$ only depend on $\{\xi_s^\tau : \tau = 1, 2, ..., t-1\}$ and $\{y_i^j(\tau,s) : \tau = 1, 2, ..., t-1\}$ and are independent of any information revealed by customer t' for $t' \geq t$. Therefore, the DWO policy is non-anticipative.

Algorithm 1 DEBT-WEIGHTED OFFER-SET (DWO) POLICY $\tilde{G}_{DWO}(s)$

Initialize: Period s, click-through goals $\boldsymbol{\alpha}(s)$, and initial debts $d_i^j(1,s) \leftarrow 0$ for all $i \in \bar{\mathcal{N}}$ and $j \in \mathcal{M}$.

For each customer $t \geq 1$**:**

1: Observe the customer-type $\xi_s^t = \hat{j}$.

2: Display the offer set $S_s^*(t)$ to the customer, where

$$S_s^*(t) \leftarrow \arg\max_{S \in \mathcal{S}_{\hat{j}}} \sum_{i \in S} \left(d_i^{\hat{j}}(t,s) \right)^+ \phi_i^{\hat{j}}(S). \tag{20}$$

3: Observe the click-throughs $(y_i^j(t,s) : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$.

4: $d_i^j(t+1,s) \leftarrow d_i^j(t,s) + \alpha_i^j(s) - y_i^j(t,s)$ for all $i \in \bar{\mathcal{N}}$ and $j \in \mathcal{M}$.

We now prove the feasibility of the debt-weighted offer-set policy under condition (13).

THEOREM 2. (FEASIBILITY OF DWO) *If the click-through goals* $\boldsymbol{\alpha}(s)$ *are feasible, i.e., inequality* (11) *or* (13) *holds for period* s*, then we have*

$$\liminf_{T_s \to +\infty} \frac{1}{T_s} \sum_{t=1}^{T_s} y_i^j(S_s^*(t)) \mathbf{1}_{\{\xi_s^t = j\}} \geq \alpha_i^j(s) \text{ with probability 1 for each } i \in \bar{\mathcal{N}} \text{ and } j \in \mathcal{M},$$

where the offer sets $\{S_s^*(t) : t \geq 1\}$ *are prescribed by the debt-weighted offer-set policy* $\tilde{G}_{DWO}(s)$*.*

We remark that the debt-weighted offer-set policy (Algorithm 1) is developed upon the theory of Blackwell's Approachability (Blackwell 1956) and online convex optimization (Hazan 2019). In a different context of resource allocation, these techniques have also been adopted by Zhong et al. (2017) and Jiang et al. (2019) to develop resource allocation algorithms to meet service-level constraints. A key difference between our approach and theirs is that we cannot fully control the behavior of the customers so that some dedicated analysis of the offer-set optimization model is required to develop the debt-weighted offer-set policy. We have now closed the loop for the analysis of the second-stage program for the offer-set optimization problem with click-through goals. Not only do we offer the necessary and sufficient feasibility condition but a feasible policy is also constructed under such a condition.

Algorithm 1 can also be implemented as a single-period randomized policy, under which, for each period s, we generate a randomized sample of $\{\xi_s^t : t \geq 1\}$ and $\{y_j^i(t,s) : t \geq 1\}$ with a sufficient large sample size, based on which the (randomized) debts and offer set for each customer type are constructed. The randomized version of the debt-weighted offer-set policy for period s (referred to as the rDWO policy, denoted as $\tilde{G}_{rDWO}(s)$) is given in Algorithm 2. Consistent with similar findings in the resource allocation literature (e.g., Zhong et al. 2017, Jiang et al. 2019), the rDWO policy will ensure the feasibility of the click-through goals $\boldsymbol{\alpha}(s)$ if (13) holds.

Algorithm 2 Randomized Debt-Weighted Offer-set (rDWO) Policy $\tilde{G}_{rDWO}(s)$

Initialize: Period s, click-through goals $\boldsymbol{\alpha}(s)$, and initial debts $d_i^j(1,s) \leftarrow 0$ for all $i \in \bar{\mathcal{N}}$ and $j \in \mathcal{M}$.

Random Sample: Randomly generate a customer-type sample $\{\xi_s^1, \xi_s^2, ..., \xi_s^{T_s}\}$ from the distribution $\mathcal{P}$, where T' is sufficiently large. For each customer-type $j \in \mathcal{M}$, denote the set of periods with $\xi_s^t = j$ as $\mathcal{T}_s^j := \{t \in [1, T_s] : \xi_s^t = j\}$.

For each customer $t = 1, 2, ..., T_s$**:**

1: Observe the customer-type $\xi_s^t = \hat{j}$.

2: Display offer set $S_s^*(t)$ to the customer, where

$$S_s^*(t) \leftarrow \arg\max_S \sum_{i \in S} \left(d_i^{\hat{j}}(t,s)\right)^+ \phi_i^{\hat{j}}(S) \tag{21}$$

3: Randomly sample customer click-throughs $(y_i^j(t,s) : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$, based on the distributions of customer click-through behaviors $\{\phi_i^j(S) : j \in \mathcal{M}, i \in \bar{\mathcal{N}}\}$.

4: $d_i^j(t+1,s) \leftarrow d_i^j(t,s) + \alpha_i^j(s) - y_i^j(t,s)$ for all $i \in \bar{\mathcal{N}}$ and $j \in \mathcal{M}$.

Output: If the realized customer-type $\xi_s = j$, then uniformly randomly pick up a time index $t \in \mathcal{T}_s^j$ and display offer set $S_s^*(t)$ defined by (21) to the customer.

Theorem 3. (Feasibility of rDWO) *If inequality* (11) *or* (13) *holds, then we have*

$$\mathbb{E}[y_i^j(S_s^j(\tilde{G}_{rDWO}(s), \xi_s))] \geq \alpha_i^j(s) \ , \ \textit{for each } i \in \bar{\mathcal{N}} \textit{ and } j \in \mathcal{M}.$$

Finally, we remark that if the customer click-through behaviors follow MNL, the DWO and rDWO algorithms can be efficiently implemented. The most time-and-space-consuming step of the algorithm to decide the offer set upon the arrival of each customer (i.e., equations (20) and (21)) can be computed easily either through solving a linear program or applying a geometric algorithm (Talluri and Van Ryzin 2004, Rusmevichientong et al. 2010, Davis et al. 2013).

5.2. Optimal Advertisement Allocation Policy

The offer-set policies introduced in Section 5.1 work well for the case with an infinite number of customers. When the platform faces a finite traffic, it is possible that the budget of some ad is depleted before the end of planning horizon. To shed light on the optimal offer-set policy with a finite number of customers, we now propose the optimal ad allocation policy based on the debt-based offer-set algorithms (Algorithms 1 and 2) and demonstrate its asymptotic optimality. Specifically, the ad allocation problem is reduced to two stages. In the first stage, i.e., click-through goal optimization, we solve the convex program (14). In the second stage, i.e., ad offer-set allocation, we adaptively customize the appropriate ad offer set to each customer upon his/her arrival.

Algorithm 3 DEBT-WEIGHTED OFFER-SET WITH BUDGET (DWO-B) POLICY $\tilde{G}_{DWO-B}$

First-stage click-through goal optimization: Solve (14) to obtain the optimal click-through goals $\boldsymbol{\alpha}^*$ and the budget $B_i^*(s)$ for all i and s (defined by Equation (23)).

Second-stage ad allocation:

For each period $s \geq 1$**:**

Initialize $d_i^j(1,s) \leftarrow 0$ for all $i \in \bar{\mathcal{N}}_s$ and $j \in \mathcal{M}$.

For each customer $t \geq 1$**:**

1: Observe the customer-type $\xi_s^t = \hat{j}$.

2: Display the offer set $S_s^*(t)$ to the customer, where

$$S_s^*(t) \leftarrow \arg\max_S \sum_{i \in S} \left(d_i^{\hat{j}}(t,s) \right)^+ \phi_i^{\hat{j}}(S). \tag{22}$$

3: Observe the customer click-throughs $(y_i^j(t,s) : i \in \bar{\mathcal{N}}_s, j \in \mathcal{M})$. The reward, $\sum_i r_i y_i^{\hat{j}}(t,s)$, is collected. In the case where the budget for ad i in period s is exhausted, i.e., $(\sum_j \sum_{\tau \leq t} y_i^j(\tau,s)) b_i \geq B_i^*(s)$, any offer set containing this ad will no longer be displayed hereafter in period s.

4: $d_i^j(t+1,s) \leftarrow d_i^j(t,s) + \alpha_i^{j*}(s) - y_i^j(t,s)$ for all $i \in \bar{\mathcal{N}}$ and $j \in \mathcal{M}$.

The DWO-B policy (Algorithm 3) first solves the optimal click-through goals $\boldsymbol{\alpha}^*$ in the presence of limited ad budgets. One should also note that (14) is a convex program so its optimal solution $\boldsymbol{\alpha}^*$ can be obtained through efficient algorithms such as gradient descent. As prescribed by Algorithm 3, our real-time offer-set strategy is to implement the DWO policy associated with the optimal click-through goals $\boldsymbol{\alpha}^*$, $\tilde{G}_{DWO}(s)$ (Algorithm 1) in each period s throughout the planning horizon. Once the budget of an ad is used up, any offer set containing this ad will not be displayed automatically under the DWO-B algorithm because the debt of this ad is no longer positive after its budget being depleted. An interesting feature of the DWO-B algorithm is that it embeds a reward maximization problem into our framework for feasible click-through goals. As a consequence, the DWO-B policy maximizes the payoff of the platform in the first stage before the campaign and adaptively displays the personalized offer set to satisfy the optimal click-through goals once the planning horizon starts. We denote

$$B_i^*(s) := \frac{B_i \zeta_s \sum_{j \in \mathcal{L}_i} \alpha_i^{j*}(s)}{\sum_{\sigma=\sigma_i}^{\sigma_i+K_i-1} \zeta_\sigma \sum_{j \in \mathcal{L}_i} \alpha_i^{j*}(\sigma)} \tag{23}$$

as the budget of ad i allocated in period s associated with the optimal click-through goals $\boldsymbol{\alpha}$.

We now demonstrate the optimality of the DWO-B policy in the asymptotic regime where the problem size scales up to infinity. Specifically, we denote a family of ad allocation problems with the budget for each ad i, $B_i(\gamma) := B_i\gamma$, the click-through requirement for ad i and customer-type set $c \subset \mathcal{L}_i$, $\eta_i^c(\gamma) = \eta_i^c\gamma$, and the planning horizon length $T(\gamma) := T\gamma$, as $\mathcal{Q}(\gamma)$, where $\gamma > 0$ is a scaling parameter of problem size. We also define $T_s(\gamma) := T_s\gamma = T\zeta_s\gamma$ $(1 \le s \le \Sigma)$ as the scaled number of customers arriving at period s, and $B_i^*(s,\gamma) = B_i^*(s)\gamma$ as the scaled budget of ad i allocated in period s. Each ad campaign i, the total number of customers during ad campaign i is $T(i,\gamma) = \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s(\gamma) = T(i)\gamma$. For problem $\mathcal{Q}(\gamma)$ and policy $\tilde{G} \in \tilde{\mathcal{G}}$, we denote $Rew(\gamma, \tilde{G})$ as its expected total reward and $Rew^*(\gamma) = \max_{\tilde{G}\in\tilde{\mathcal{G}}} Rew(\gamma, \tilde{G})$ as the optimal expected reward of $\mathcal{Q}(\gamma)$.

LEMMA 1. *For problem $\mathcal{Q}(\gamma)$, Algorithm 3 is feasible with respect to the optimal click-through goals $\boldsymbol{\alpha}^*$ as $\gamma \uparrow +\infty$, i.e.,*

$$\liminf_{\gamma\to+\infty} \frac{1}{T_s(\gamma)} \sum_{t=1}^{T_s(\gamma)} y_i^j(t,s) \ge \alpha_i^{j*}(s) \text{ for each } i \in \bar{\mathcal{N}}_s,\ j \in \mathcal{M}, \text{ and } s.$$

Adopting a coupling argument, the proof of Lemma 1 (see Appendix C for details) shows that if the problem size γ scales up to infinity, the DWO-B policy will *not* exhaust the budget of any ad and will secure the first-stage optimal click-through goals $\boldsymbol{\alpha}^*(s)$ of each period s. As a consequence, the click-through requirements $\{\eta_i^c(\gamma) : i \in \mathcal{N}, c \subset \mathcal{L}_i\}$ can be satisfied as well. Lemma 1 paves our way to establish the asymptotic optimality of the DWO-B policy in the following theorem.

THEOREM 4. *The debt-weighted offer-set with budget (DWO-B) policy is asymptotically optimal, i.e.,*

$$\lim_{\gamma\to+\infty} \frac{Rew(\gamma, \tilde{G}_{DWO-B})}{Rew^*(\gamma)} = 1 \tag{24}$$

Theorem 4 proves that the DWO-B policy is optimal when the ad budget, the click-through requirement, and then time horizon length scales up to infinity at the same rate. The proof of Theorem 4 also implies that the optimality gap of the DWO-B algorithm is of order $\mathcal{O}(\sqrt{\gamma})$. Hence, even if the problem scale parameter γ is small, this algorithm could still generate impressive reward performances for the platform, as shown by our numerical experiments in Section 6.

A classical approach in the literature is to formulate the offer-set optimization problem in the presence of ad budget constraints as a linear program (called choice-based linear program, CBLP, see Liu and Van Ryzin 2008). See the linear program $\mathcal{LP}(\gamma)$ in Appendix C. Throughout each period, the firm adopts a (stationary) randomized offer-set policy based on the solution to the CBLP. Hence, the CBLP approach is not adaptive, which will result in substantial variability in both the reward performance and the budget depletion process. Our proposed DWO-B algorithm, however, responds to the randomness in customer types and click-through behaviors. That is, if the (realized) per-customer click-throughs of ad i from type-j customers in period s do not reach the optimal click-through goal $\alpha_i^{j*}(s)$, ad i will be more likely to be included in the offer set displayed to the next type j customer. In other words, the offer sets displayed to customers under the DWO-B algorithm are correlated, so that the debt process exhibits a mean-reverting pattern. As a consequence, although both CBLP and DWO-B algorithms achieve the asymptotic optimality, the latter policy generates significantly less variable reward and budget streams, and a higher expected reward when the loading factor is high (i.e., when demand exceeds supply) in the non-asymptotic regime, as shown by our numerical experiments in Section 6.

It is worthwhile noting that the DWO-B algorithm may be too conservative at the beginning of an ad campaign because it places too much emphasis on meeting the (optimal) click-through goals instead of maximizing the (immediate) reward. To address this issue, we consider a variant of the DWO-B algorithm, which uses the optimal expected *total reward* of each ad (irrespective of customer type) as the target to construct the debt process. This new algorithm, on the one hand, achieves the same asymptotic optimality (and convergence rate) as the DWO-B algorithm, and, on the other hand, performs better when the customer traffic T is low because it generates a higher reward at the beginning of an ad campaign. We present the offer-set policy with aggregate reward targets in Algorithm 4, and an analog with individualized reward targets in Algorithm 6 in Appendix D.

By changing the personalized click-through goals of ad-customer pairs to the aggregate reward target of each ad, the DWO-ART policy (Algorithm 4) directly controls the total reward from each ad to optimality. As we show in the following proposition, this algorithm preserves the asymptotic optimality property.

PROPOSITION 3. *The debt-weighted offer-set with aggregate reward target (DWO-ART) policy is asymptotically optimal, i.e.,*

$$\lim_{\gamma \to +\infty} \frac{Rew(\gamma, \tilde{G}_{DWO-ART})}{Rew^*(\gamma)} = 1 \tag{25}$$

The DWO-ART algorithm further reduces the variability of the realized reward process by aggregating together the reward to an ad level. As a consequence, a key advantage of DWO-ART over DWO-B is that the former achieves a higher reward performance in general, especially at the beginning of the ad campaigns, since it directly operates the debt on the aggregate reward of each ad. This is an appealing feature when implementing the algorithm in practice where a stable reward process is preferred. Finally, we remark that the DWO-ART algorithm guarantees the click-through requirements $(\eta_i^c(\gamma) : i \in \bar{\mathcal{N}}, c \subset \mathcal{L}_i)$, as long as the optimal click-through goal vector $\boldsymbol{\alpha}^*$ is unique. In the case where $\boldsymbol{\alpha}^*$ is not unique, our numerical experiments show that the required click-through targets $((\eta_i^c(\gamma): i \in \bar{\mathcal{N}}, c \subset \mathcal{L}_i)$, if any) can be satisfied under DWO-ART for all of the numerical instances we examined.

Algorithm 4 DEBT-WEIGHTED OFFER-SET WITH AGGREGATE REWARD TARGET (DWO-ART) POLICY $\tilde{G}_{DWO-ART}$

First-stage click-through goal optimization: Solve (14) to obtain the optimal click-through goals $\boldsymbol{\alpha}^*$ and the budget $B_i^*(s)$ for all i and s.

Second-stage ad allocation:

For each period $s \geq 1$**:**

Initialize $d_i(1, s) \leftarrow 0$ for all $i \in \bar{\mathcal{N}}$.

For each customer $t \geq 1$**:**

1: Observe the customer-type $\xi_s^t = \hat{j}$.

2: Display offer set $S_s^*(t)$ to the customer, where

$$S_s^*(t) \leftarrow \arg\max_S \sum_{i \in S} (d_i(t,s))^+ \phi_i^{\hat{j}}(S). \tag{26}$$

3: Observe the customer click-throughs $(y_i^j(t,s): i \in \bar{\mathcal{N}}, j \in \mathcal{M})$. The reward, $\sum_i r_i y_i^{\hat{j}}(t,s)$, is collected. In the case where the budget for ad i in period s is exhausted, i.e., $(\sum_j \sum_{\tau \leq t} y_i^j(\tau, s)) b_i \geq B_i^*(s)$, any offer set containing this ad will no longer be displayed hereafter in period s.

4: $d_i(t+1, s) \leftarrow d_i(t,s) + \sum_{j \in \mathcal{M}} r_i \alpha_i^{j*}(s) - \sum_{j \in \mathcal{M}} r_i y_i^j(t,s)$ for all $i \in \bar{\mathcal{N}}$.

6. Numerical Experiments

In this section, we numerically evaluate the DWO-B and DWO-ART algorithms for ad allocation optimization, benchmarked against the well-established choice-based linear program (CBLP) method and the LP resolving heuristics. The key takeaways from our numerical experiments are that (1) thanks to the adaptive offer set personalization, our algorithm achieves better reward performances especially when the load factor is large; (2) the rewards generated by DWO-B and DWO-ART policies are much more stable (i.e., with lower variability in the realized reward) than those of CBLP and LP-resolving heuristics because the debt process directly steers the click-through goals of each ad from each customer segment toward the respective optimal click-through goal throughout the planning horizon.

Our numerical setup is summarized as follows. We consider a planning horizon (1 day) with 4 periods (6 hours each) and 125 ad campaigns to serve 5 customer segments. We sample the per-click value of each ad $\{r_1, r_2, ..., r_{125}\}$ independently from a uniform distribution on the interval $[10, 50]$. For all campaigns, we set the starting active period $\sigma_{5k+1} = 1$, $\sigma_{5k+2} = 3$, $\sigma_{5k+3} = 1$, $\sigma_{5k+4} = 2$, and $\sigma_{5k+5} = 4$ and the active length $K_{5k+1} = 2$, $K_{5k+2} = 2$, $K_{5k+3} = 1$, $K_{5k+4} = 2$, and $K_{5k+5} = 1$ for $k = 0, 1, ..., 24$. Then in each period s, the number of active ad campaigns is $n_s := |\mathcal{N}_s| = 50$. We assume each period has a different number of arriving customers with $T_1 = 500$, $T_2 = 1{,}000$, $T_3 = 1{,}000$, and $T_4 = 1{,}500$ (so the total number of customers is $T = 4{,}000$). The budget B_i of each ad campaign is the sum of potential budgets in its active periods, which we set as proportional to $r_i T_s / n_s$ in different periods. The customer-type distribution $(p_1^s, p_2^s, ..., p_5^s)$, where $\sum_j p_j^s = 1$, is generated from a 5-dimension Dirichlet distribution in each period s.

We model the click-through behaviors of the customers using MNL, i.e., for $i \in S \subset \mathcal{S}_j$,

$$\phi_i^j(S) = \frac{v_i^j}{1 + \sum_{i' \in S} v_{i'}^j}.$$

Each ad-customer pair is associated with an attraction index v_i^j. For ad i and customer-type j, we assign $v_i^j := \exp(u_i^j)$, where the expected utility of customer-type j to ad i, u_i^j, is independently sampled from the uniform distribution on the interval $[-1, 2]$. In the basic setting of our numerical studies, we set the cardinality constraint of an offer set as 3, i.e., $|S_s^t| \leq 3$ for each s and t. Finally, we remark that the above parameter combinations are set without loss of generality. All our results and insights presented in this section are robust with respect to different values.

To uncover insights on when our debt-based algorithms will be most valuable relative to well-established LP-based benchmarks, we systematically vary two focal parameters: (a) the concentration parameter (CP) associated with the proportion of each customer type, and (b) the loading factor (LF), defined as the ratio between the total expected demand and total supply. Specifically, the concentration parameters are determined by the parameters of the Dirichlet distribution we use to generate $(p_1, p_2, \cdots, p_5)$. The loading factor is the ratio between the total user traffic and the total affordable traffic with the budgets of the advertisers. See Appendix B for detailed explanations of the CP parameter. As is clear from their definitions, CP measures the uniformness of the customer-type distribution and LF measures the tightness of the ad budget. The higher the CP, the more uniform the distribution of customer types; the higher the LF, the tighter the budget constraint for the ad campaigns. In our experiments, we vary CP from 0.1 to 100, and LF from 1 to 1.8.

We consider the CBLP approach and its resolving variation as the benchmarks to evaluate our debt-based algorithms. Specifically, let $z_s^j(S)$ denote the probability of offer set S to be shown to a customer of type j in period s. Then, we can formulate the offer-set decision of the firm throughout the planning horizon as the following linear program:

$$
\begin{aligned}
\max \quad & \sum_{s,i,j,S\in\mathcal{S}_j} T_s r_i p_j^s \phi_i^j(S) z_s^j(S) \\
s.t. \quad & \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s b_i \sum_{j,S} p_j^s \phi_i^j(S) z_s^j(S) \le B_i \text{ for all } i \in \mathcal{N} \\
& \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s \sum_{j\in c, S\in\mathcal{S}_j} p_j^s \phi_i^j(S) z_s^j(S) \ge \eta_i^c \text{ for all } i \in \bar{\mathcal{N}} \text{ and } c \subset \mathcal{M} \\
& \sum_{S\in\mathcal{S}_j} z_s^j(S) \le 1 \text{ for all } j \in \mathcal{M} \text{ and } s \\
& z_s^j(S) \ge 0 \text{ for all } j \in \mathcal{M}, S \in \mathcal{S}_j, \text{ and } s
\end{aligned}
\tag{27}
$$

Compared with the standard CBLP approach, the LP formulation (27) has an additional constraint that the total sales of advertisement i to customer-segment set c should meet the corresponding click-through target. Based on the solution to (27), one can construct a randomized algorithm under which, upon the arrival of a type j customer, the offer-set S is randomly provided to the customer with probability $z_s^{j*}(S)$. Note that (27) involves $\mathcal{O}(m2^n)$ decision variables and $\mathcal{O}(n2^m)$ constraints and is thus computationally intractable if the number of ads n or the number of customer segments m is large.

On top of the CBLP approach, we also adopt an LP-resolving heuristic as another benchmark (Jasin and Kumar 2012). Specifically, at the beginning each period s, we resolve the CBLP based on the remaining budget of each ad, remaining time periods $\Sigma - s + 1$, and unfinished click-through targets in place of the initial budget B_i, total time periods Σ, and initial click-through targets $(\eta_i^c : i \in \mathcal{N}, c \subseteq \mathcal{L}_i)$. The firm then adopts the randomized algorithm induced by the solution to the resolving LP until the beginning of the next period. In our numerical studies, we resolve the LP for 4 times in total. In the case where one resolving LP is infeasible, we will keep the randomized policy of the previous resolving epoch unchanged. In both CBLP and LP-resolving approaches, if an advertisement is out of budget before the end of the planning horizon, we remove this ad from the offer set generated by the algorithm until the end of the horizon.

We report our numerical findings in Table 1, with the ratio between the standard error of the total reward to the theoretical upper-bound included in the bracket. It is clear from our experiments that the proposed debt-based algorithms achieve near-optimal reward performances in the nonasymptotic regime. We highlight the following takeaways. First, although the CBLP and LP-resolving approaches could already achieve reward performances close to the theoretical upper bound, the proposed debt-based algorithms could generate even higher rewards especially when the loading factor (LF) is high. We emphasize that the CBLP and LP-resolving benchmarks are nonadaptive so that the firm does not adjust the probability of each offer set in real time. When the total budget is much lower than the

CP	LF	LP	LP-Resolving	DWO-B	DWO-ART
0.1	1.8	92.01% (1.06%)	94.06% (1.24%)	99.92% (0.07%)	100.00% (0.00%)
	1.5	93.16% (0.82%)	94.37% (0.75%)	99.57% (0.36%)	100.00% (0.00%)
	1.2	94.02% (0.80%)	95.28% (0.73%)	99.34% (0.41%)	100.00% (0.00%)
	1.0	95.23% (0.51%)	95.78% (0.49%)	98.52% (0.27%)	98.19% (0.37%)
1	1.8	92.09% (1.05%)	94.33% (1.03%)	99.91% (0.08%)	100.00% (0.00%)
	1.5	92.91% (0.77%)	95.09% (0.66%)	99.87% (0.17%)	100.00% (0.00%)
	1.2	94.17% (0.72%)	95.52% (0.60%)	98.50% (0.57%)	100.00% (0.00%)
	1.0	95.37% (0.54%)	96.28% (0.48%)	98.60% (0.31%)	97.62% (0.27%)
10	1.8	91.81% (1.08%)	94.54% (0.79%)	100.00% (0.00%)	100.00% (0.00%)
	1.5	93.24% (0.89%)	95.15% (0.85%)	100.00% (0.01%)	100.00% (0.01%)
	1.2	94.07% (0.73%)	95.31% (0.82%)	99.83% (0.23%)	100.00% (0.00%)
	1.0	95.38% (0.56%)	96.35% (0.43%)	98.77% (0.27%)	97.73% (0.27%)
100	1.8	92.14% (0.71%)	94.39% (1.16%)	100.00% (0.00%)	100.00% (0.00%)
	1.5	92.92% (0.72%)	95.43% (0.58%)	100.00% (0.00%)	100.00% (0.00%)
	1.2	93.83% (0.69%)	95.39% (0.63%)	98.73% (0.72%)	100.00% (0.00%)
	1.0	95.25% (0.68%)	96.16% (0.54%)	98.78% (0.27%)	97.64% (0.25%)

Table 1 Numerical Results (Standard Error relative to the Theoretical Upper-bound in Parentheses)

total customer demand (i.e., LF is very high), our debt-based algorithms outperform the LP-based heuristics. In this case, our proposed policies aim to clear all the budget, which proves to dominate the randomized (stationary) offer-set policy generated by LP-based heuristics given that demand far exceeds supply. When LF is low, however, the demand side is the bottleneck, so that the LP-based heuristics that exploit the user traffic to generate the highest value may become more competitive. The second important insight from our numerical analysis is that, the debt-based DWO-B and DWO-ART algorithms generate much more stable rewards (i.e., much lower standard deviations) than CBLP and LP-resolving heuristics. To understand the underlying intuition of this result, we note that the debt-based algorithms assign a higher weight (i.e., debt) to the advertisement/customer-type pair that is farther away from the optimal click-through goal to autonomously drive the system to optimality. As a consequence, the value (and, thus, budget depletion) trajectory would follow a mean-reverting pattern under the DWO-B and DWO-ART algorithms with lower variability than under the LP-based benchmarks.

To further demonstrate the low variability of realized reward under our two policies, we also show quantiles of realized click-throughs of the highest-value advertisement: 0.1 quantile, median and 0.9 quantile, changing over time under the four approaches studied (with $LF = 1$ and $CP = 100$) in Figure 3. We consider the ad with the highest reward associated with 30 sample paths. Although all four approaches deplete the ad's entire budget for more than 50 percent of sample paths, the variability of realized click-throughs (i.e., budget depleting process) through entire time horizon under CBLP and LP-resolving algorithms are much higher than our DWO-B and DWO-ART policies. There are nonnegligible chances of running out the budget well in advance for the ad with the highest reward under CBLP and LP-resolving algorithms, while the budget is only exhausted toward the very end of the time horizon for our two policies. Such smooth budget depletion processes of our proposed algorithms should be credited to their mean-reverting pattern.

To highlight the mean-reverting property of our proposed algorithms, we examine the intertemporal correlations of the *per-period debt*, defined as $\Delta_i^j(t,s) := d_i^j(t,s)/t$ where $d_i^j(t,s)$ is the debt of ad i for customer segment j in period s after t customers arriving at the platform (as defined in Algorithm 3). More specifically, for each of the 4 algorithms studied in our numerical experiments, we regress the per-period debt in the next period on the per-period debt in the current period using the following model specification:

$$\Delta_i^j(t+1,s) = a_0 + a_1 \Delta_i^j(t,s) + \epsilon$$

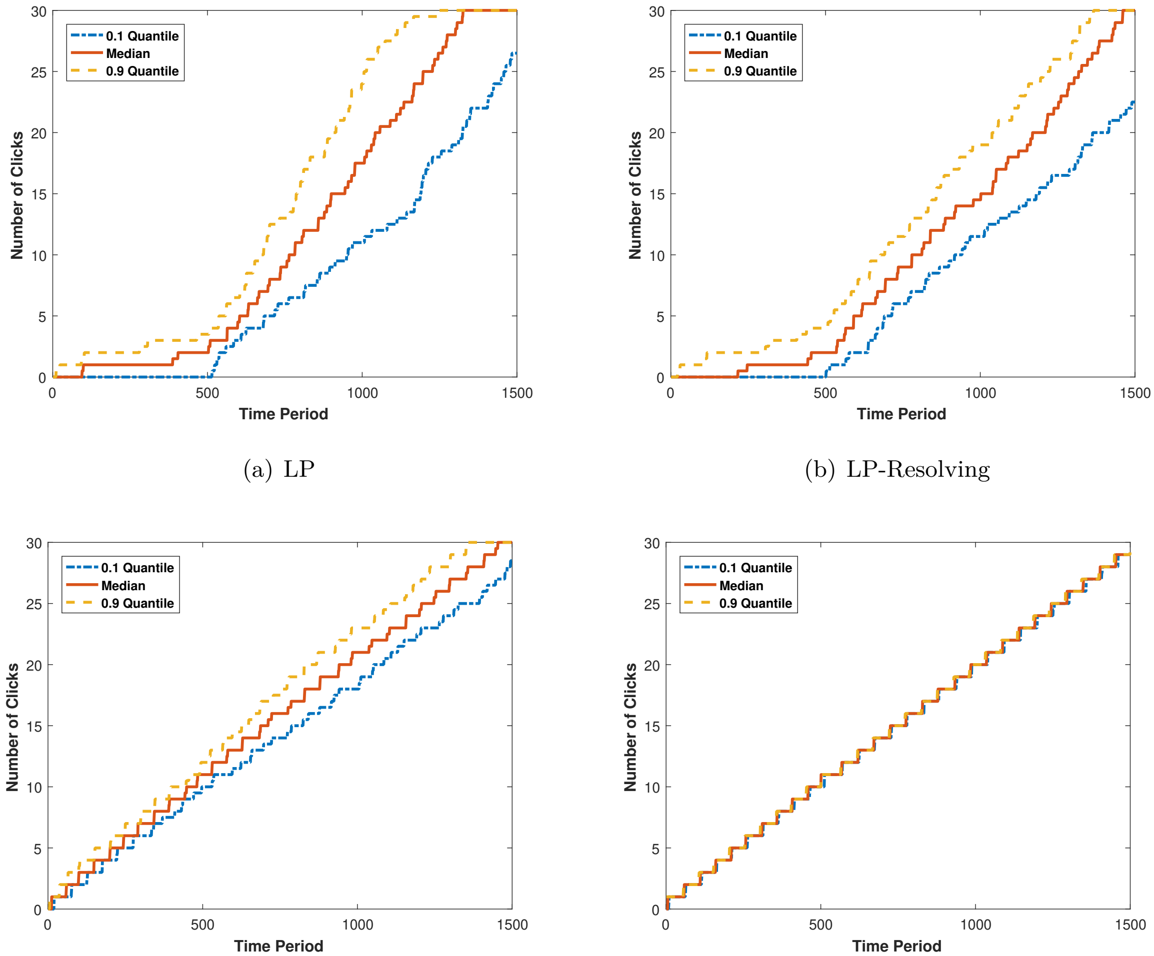
The regression results are reported in Table 2.

Approach	Coefficient	Estimation	Standard Error	t-statistics	p-value
LP	a_0	-5.0109e-05	2.2898e-06	-21.884	3.8697e-106
	a_1	0.96504	0.00016274	5929.9	0
LP-Resolving	a_0	-6.7653e-05	2.4747e-06	-27.338	1.6947e-164
	a_1	0.95897	0.00016962	5653.8	0
DWO-B	a_0	-0.00011368	1.8686e-06	-60.835	0
	a_1	0.86632	0.00044006	1968.7	0
DWO-ART	a_0	-0.00042431	4.0243e-06	-105.44	0
	a_1	0.64191	0.00017748	3616.9	0

Table 2 The Regression Results of the Intertemporal Correlations of Per-Period Debt

As shown in Table 2, the per-period debt processes exhibit the mean-reverting pattern for our proposed DWO-B and DWO-ART algorithms, captured by the fact that the estimate $\hat{a}_1$ is positive and statistically significantly below 1 for these policies. An important observation from our regression results is that, the estimated coefficient $\hat{a}_1$ is substantially lower for our algorithms (0.866 for DWO-B and 0.642 for DOW-ART) than that for the LP-based benchmarks (0.965 for CBLP and 0.959 for LP-Resolving). Such an observation delivers an intriguing insight that our debt-based algorithms drive the per-period debt process towards its mean and, as a consequence, result in a more stable budget depletion process for the ads.

In addition to obtaining the better performance in most of the cases we examine and much smoother depletion of the budget, our DWO-B and DWO-ART approaches are more scalable and efficient in both the time and space complexities. We carried out our numerical studies by varying the offer-set size constraint K from 2 to 6. The experiment is conducted by using Gurobi 9.1 within MATLAB



(a) LP

(b) LP-Resolving

(c) DWO-B

(d) DWO-ART

Figure 3 The 0.1 Quantiles, Medians, and 0.9 Quantiles in 30 Sample Paths over Time of Click-Numbers of the Highest-Reward Advertisement with $LF = 1$, and $CP = 100$

9.0 on a 2.20GHz Intel Core i5-5200 CPU with 8.00 GB of RAM. As shown in Table 3, the average computation time of finding optimal click-through goals (i.e., solving the convex program (14)) is approximately 0.01s regardless of the value of K, but the CBLP (27) approach is much more time-consuming, so is the case with the LP resolving heuristic. In addition, increasing K means more possible offer sets in the LP and LP re-solving formulations, so a larger memory is necessary in this case. As shown in Table 3, the case of $K = 6$ incurs an "out of memory" error for the CBLP benchmark. In short, our algorithms enjoy higher scalability than the LP-based benchmarks.

7. Conclusion

The allocation of customer traffic to different ads is a crucial operations decision for an online e-commerce platform to optimize its advertising business. This paper proposes a general modeling

$n=50, m=5$	$K=2$	$K=3$	$K=4$	$K=5$	$K=6$
Our approach	0.011s	0.013s	0.008s	0.008s	0.010s
CBLP	0.002s	0.058s	1.13s	15.98s	out of memory

Table 3 The Comparison of Average Solving Time

framework and the associated efficient algorithms to study optimal ad allocation in the presence of customer choices and limited advertising budgets. We offer several interesting takeaways.

We propose a general two-stage stochastic program modeling framework to study the optimal online ad allocation. Our framework is flexible and parsimonious, enabling us to reduce the complex stochastic program into a simple convex optimization model to obtain the optimal click-through goals. Associated with the proposed framework, we develop a family of simple and effective algorithms, referred to as the debt-weighted offer-set policy. This policy is provably optimal for the ad allocation problem in the asymptotic regime with a large problem size. This policy assigns higher weights to the ad-customer pairs that have larger "debts" from the preset click-through goals and optimizes the offer-set decisions accordingly. The debts serve as a simple but efficient mechanism that synchronizes the first-stage problem to maximize the total value and the second stage to meet the click-through goals.

Our modeling framework enables us to further reveal interesting insights for the ad allocation problem. Through numerical experiments, we demonstrate that our algorithm outperforms the well-studied CBLP policy and the LP-resolving heuristic especially when the loading factor is high. Furthermore, the proposed algorithms give rise to the mean-reverting pattern of the per-period debt process and, therefore, achieve smoother budget depletion processes, which are very much desirable from a practical perspective.

While this paper provides some answers on how online advertising platforms can optimize an ad allocation policy in the presence of customer choices, it raises several intriguing questions as well. First, our algorithm assumes that the planning horizon length (i.e., the total number of customers) is known apriori. This assumption may not be valid for the ads that have an uncertain total customer traffic. It is interesting to strengthen our approach to address the problem with a random total number of customers. Second, in our framework, the click-through model is assumed to be known to the platform. It is an exciting direction to explicitly model how the platform adaptively learns the click-through distribution of each customer type from the data of sequentially arriving customers. We leave these questions for future research.

References

Adler, Micah, Phillip B Gibbons, Yossi Matias. 2002. Scheduling space-sharing for internet advertising. *Journal of Scheduling* **5**(2) 103–119.

Alptekinoğlu, Aydın, Arunava Banerjee, Anand Paul, Nikhil Jain. 2013. Inventory pooling to deliver differentiated service. *Manufacturing & Service Operations Management* **15**(1) 33–44.

Balseiro, Santiago R, Jon Feldman, Vahab Mirrokni, Shan Muthukrishnan. 2014. Yield optimization of display advertising with ad exchange. *Management Science* **60**(12) 2886–2907.

Bernstein, Fernando, Sajad Modaresi, Denis Sauré. 2019. A dynamic clustering approach to data-driven assortment personalization. *Management Science* **65**(5) 2095–2115.

Blackwell, David. 1956. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics* **6**(1) 1–8.

Caro, Felipe, Victor Martínez-de Albéniz, Paat Rusmevichientong. 2014. The assortment packing problem: Multiperiod assortment planning for short-lived products. *Management Science* **60**(11) 2701–2721.

Chen, Xi, Will Ma, David Simchi-Levi, Linwei Xin. 2016. Assortment planning for recommendations at checkout under inventory constraints. *Available at SSRN 2853093* .

Cheung, Wang Chi, David Simchi-Levi. 2017. Thompson sampling for online personalized assortment optimization problems with multinomial logit choice models. *Available at SSRN 3075658* .

Chickering, David Maxwell, David Heckerman. 2003. Targeted advertising on the web with inventory management. *Interfaces* **33**(5) 71–77.

Choi, Hana, Carl F Mela, Santiago R Balseiro, Adam Leary. 2020. Online display advertising markets: A literature review and future directions. *Information Systems Research* .

Davis, James, Guillermo Gallego, Huseyin Topaloglu. 2013. Assortment planning under the multinomial logit model with totally unimodular constraint structures. *Work in Progress* .

Dong, Lingxiu, Duo Shi, Fuqiang Zhang. 2017. 3D printing vs. traditional flexible technology: Implications for operations strategy. *Available at SSRN 2847731* .

Eppen, Gary D. 1979. Note—effects of centralization on expected costs in a multi-location newsboy problem. *Management science* **25**(5) 498–501.

Feldman, Jacob, Dennis Zhang, Xiaofei Liu, Nannan Zhang. 2018. Taking assortment optimization from theory to practice: Evidence from large field experiments on alibaba. Working Paper.

Feldman, Jon, Nitish Korula, Vahab Mirrokni, Shanmugavelayutham Muthukrishnan, Martin Pál. 2009. Online ad assignment with free disposal. *International Workshop on Internet and Network Economics*. Springer, 374–385.

Gallego, Guillermo, Anran Li, Van-Anh Truong, Xinshang Wang. 2016. Online personalized resource allocation with customer choice. Tech. rep., Working Paper. http://arxiv. org/abs/1511.01837 v1.

Golrezaei, Negin, Hamid Nazerzadeh, Paat Rusmevichientong. 2014. Real-time optimization of personalized assortments. *Management Science* **60**(6) 1532–1551.

Hao, Xiaotian, Zhaoqing Peng, Yi Ma, Guan Wang, Junqi Jin, Jianye Hao, Shan Chen, Rongquan Bai, Mingzhou Xie, Miao Xu, Zhenzhe Zheng, Chuan Yu, HAN LI, Jian Xu, Kun Gai. 2020. Dynamic knapsack optimization towards efficient multi-channel sequential advertising. *ICML 2020* .

Hazan, Elad. 2019. Introduction to online convex optimization. *arXiv preprint arXiv:1909.05207* .

Hou, I-Hong, Vivek Borkar, PR Kumar. 2009. *A theory of QoS for wireless*. IEEE.

Jasin, Stefanus, Sunil Kumar. 2012. A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research* **37**(2) 313–345.

Jiang, Jiashuo, Shixin Wang, Jiawei Zhang. 2019. Achieving high individual service-levels without safety stock? optimal rationing policy of pooled resources. Working Paper.

Kallus, Nathan, Madeleine Udell. 2020. Dynamic assortment personalization in high dimensions. *Operations Research* .

Kumar, Subodha, Varghese S Jacob, Chelliah Sriskandarajah. 2006. Scheduling advertisements on a web page to maximize revenue. *European Journal of Operational Research* **173**(3) 1067–1089.

Liu, Qian, Garrett Van Ryzin. 2008. On the choice-based linear programming model for network revenue management. *Manufacturing & Service Operations Management* **10**(2) 288–310.

Lyu, Guodong, Wang-Chi Cheung, Mabel C Chou, Chung-Piaw Teo, Zhichao Zheng, Yuanguang Zhong. 2019. Capacity allocation in flexible production networks: Theory and applications. *Management Science* **65**(11) 5091–5109.

Lyu, Guodong, Mabel C Chou, Chung-Piaw Teo, Zhichao Zheng, Yuanguang Zhong. 2017. Capacity allocation with stock-out probability targets: Theory and applications. Working Paper.

Ma, Will, Pan Xu. 2020. Group-level fairness maximization in online bipartite matching. *arXiv preprint arXiv:2011.13908* .

Microsoft. 2020. *Partner Incentives Co-op Guidebook: Business Policies for FY21*. Microsoft Corporation.

Rusmevichientong, Paat, Zuo-Jun Max Shen, David B Shmoys. 2010. Dynamic assortment optimization with a multinomial logit choice model and capacity constraint. *Operations research* **58**(6) 1666–1680.

Shen, Huaxiao, Yanzhi David Li, Youhua Frank Chen, Kai Pan. 2020. Integrated ad delivery planning for targeted display advertising. *Operations Research* .

Swaminathan, Jayashankar M, Ramesh Srinivasan. 1999. Managing individual customer service constraints under stochastic demand. *Operations Research Letters* **24**(3) 115–125.

Talluri, Kalyan, Garrett Van Ryzin. 2004. Revenue management under a general discrete choice model of consumer behavior. *Management Science* **50**(1) 15–33.

Turner, John. 2012. The planning of guaranteed targeted display advertising. *Operations research* **60**(1) 18–33.

Walsh, William E, Craig Boutilier, Tuomas Sandholm, Rob Shields, George Nemhauser, David C Parkes. 2010. Automated channel abstraction for advertising auctions. *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. 887–894.

Wang, Ruxian. 2012. Capacitated assortment and price optimization under the multinomial logit model. *Operations Research Letters* **40**(6) 492–497.

Yang, Jian, Erik Vee, Sergei Vassilvitskii, John Tomlin, Jayavel Shanmugasundaram, Tasos Anastasakos, Oliver Kennedy. 2012. Inventory allocation for online graphical display advertising using multi-objective optimization. *ICORES* **12** 293–304.

Ye, Zikun, Dennis Zhang, Heng Zhang, Renyu Philip Zhang, Xin Chen, Zhiwei Xu. 2020. Cold start on online advertising platforms: Data-driven algorithms and field experiments. *Available at SSRN* .

Zhong, Yuanguang, Zhichao Zheng, Mabel C Chou, Chung-Piaw Teo. 2017. Resource pooling and allocation policies to deliver differentiated service. *Management Science* **64**(4) 1555–1573.

Online Appendices

Appendix A. Summary of Notations

Notation	Description
Σ	Number of periods in the planning horizon
T (T_s)	Number of customers in a day (in period s)
η_s	Fraction of customers in period s
$\mathcal{N}$ ($\mathcal{N}_s$)	Set of (active) ad campaigns (in period s)
n	Number of ad campaigns
$\bar{\mathcal{N}}$ ($\bar{\mathcal{N}}_s$)	Set of all ad campaigns and the organic recommendation in a day (in period s)
λ_i	vector of parameters of ad campaign i
σ_i	Starting period of ad campaign i
K_i	Length of periods of ad campaign i
B_i	Total budget of ad campaign i
b_i	Bid price of ad campaign i set by advertiser
$\mathcal{T}_s$	s-period horizon
ξ_s^t	Customer-t type in period s
$\mathcal{M}$	Set of customer types
m	Number of customer types
p_j^s	Probability of a customer being type j in period s
S (S_s^t)	Offer set (for customer t in period s)
$y_i^j(S)$	Total number of click-throughs of a type-j customer on ad campaign i, given offer set S
$\phi_i^j(S)$	Expected value of $y_i^j(S)$
$\mathcal{L}_i$	Set of customer types targeted by ad campaign i
η_i^c	Required click-throughs for customer-type set c on ad campaign i
$\mathcal{S}_j$	Collection of all possible offer sets for type-j customers
r_i	Value of each click of ad campaign i
$\alpha_j^i(s)$	Per-user number of click-throughs for ad campaign i from type-j customers in period s
$\boldsymbol{\alpha}$	Set of $\alpha_j^i(s)$
$\tilde{G}$	Randomized offer set policy
$\tilde{\mathcal{G}}$	Set of all feasible offer set policy
$\mathcal{P}$	Distribution of customer type
K	Maximum size of an offer set
x_i^j	Indicator of offering adcampaign i to type-j customers
$\mu_j(S)$	Probability of displaying offer-set S to type-j customers
v_i^j	Attractiveness of ad campaign i to type-j customers
$d_i^j(t,s)$	Debt of type-j customer t on ad campaign i in period s
γ	Scaling parameter of problem size
$\mathcal{Q}(\gamma)$	Family of ad allocation problems given scaling parameter γ
$Rew(\gamma,\bar{G})$	Expected total reward of $\mathcal{Q}(\gamma)$
u_i^j	Expected utility of ad campaign i to type-j customers
$N^j(S)$	Total number of periods where S is offered to type-j customers.

Table A.1 Summary of Notations

Appendix B. Concentration Parameter

In our numerical experiments (Section 6), we vary CP to change the uniformness of proportions $p_1, ..., p_m$ of m customer types, which are generated by a m-dimension Dirichlet distribution. The m-dimension Dirichlet distribution has m concentration parameters $\beta_1, ..., \beta_m$. In our experiments, we set $CP := \beta_0 = \beta_1 = \beta_2 = ... = \beta_m$. Note that, for all j, $\mathbb{E}[p_j] = \frac{\beta_j}{\sum_{k=1}^m \beta_k} = \frac{1}{m}$ and $Var(p_j) = \frac{m-1}{m^2(m\beta_0+1)}$, which is decreasing in β_0. For $j \neq k$, the covariance between p_j and p_k is $-\frac{1}{m^2(m\beta_0+1)}$, which is increasing in β_0. Hence, if β_0 is larger, the sampled customer type distribution will be close to the uniform distribution on $\{1, 2, ..., m\}$. In contrast, if $CP = \beta_0$ is small, the customer type distribution is more likely to be concentrated on a subset of $\{1, 2, ..., m\}$. In other words, the higher the CP, the more uniform the generated distribution of customer types.

Appendix C. Proofs of Statements

Proof of Theorem 1

The feasibility of (4) is equivalent to that of (8). By strong duality and (10), we have (8) is feasible only if

$$\min_{\theta_i^j \geq 0} \left\{ \max_{G_s \in \mathcal{G}_s} \sum_{i \in \bar{\mathcal{N}}, j \in \mathcal{M}} p_j^s \phi_i^j(S_s^j(G_s, j))\theta_i^j - \sum_{i \in \bar{\mathcal{N}}, j \in \mathcal{M}} \alpha_i^j(s)\theta_i^j \right\} \geq 0,$$

which is equivalent to (11). This completes the proof of Theorem 1. □

Proof of Corollary 1

The proof follows immediately from Theorem 1 and Equation (15). □

Proof of Proposition 1

A standard result in the assortment optimization literature postulates that the left-hand side of (16) is quasi-convex in $x(s)$, so there always exists a maximizer on the boundary of the feasible region. Thus, we can relax the binary constraint $x_i^j(s) \in \{0, 1\}$ to $x_i^j(s) \in [0, 1]$ in (16), which is therefore equivalent to

$$\max_{x_i^j(s) \in [0,1], \sum_i x_i^j(s) \leq K} \sum_{i \in \bar{\mathcal{N}}, j \in \mathcal{M}} \frac{p_j^s v_i^j \theta_i^j x_i^j(s)}{1 + \sum_{i' \in \bar{\mathcal{N}}} v_{i'}^j x_{i'}^j(s)} \geq \sum_{i \in \bar{\mathcal{N}}, j \in \mathcal{M}} \alpha_i^j(s)\theta_i^j \text{ for all } \theta_i^j \geq 0 \ (i \in \bar{\mathcal{N}}, j \in \mathcal{M}) \tag{28}$$

We change the decision variable and define

$$z^j := 1/(1 + \sum_{i'} v_{i'}^j x_{i'}^j(s)) \text{ and } y_i^j := x_i^j(s) z^j = x_i^j(s)/(1 + \sum_{i'} v_{i'}^j x_{i'}^j(s)).$$

Since $x_0^j(s) = 1$ for all j, we have $y_0^j = z^j$ for all $j \in \mathcal{M}$. Then, we can rewrite (28) as

$$\begin{aligned}
&\min_{\theta_i^j \geq 0} \left(\max_{y_i^j, z^j} \sum_{i \in \mathcal{N}, j \in \mathcal{M}} p_j^s v_i^j y_i^j \theta_i^j - \sum_{i \in \mathcal{N}, j \in \mathcal{M}} \alpha_i^j(s)\theta_i^j \right) \geq 0 \\
&s.t. \ \sum_{i=0}^n v_i^j y_i^j + z^j = 1, \\
&\qquad \sum_{i \in \bar{\mathcal{N}}} y_i^j \leq K z^j, \text{ for all } j \in \mathcal{M}, \\
&\qquad y_i^j \leq z^j \text{ for all } i \in \bar{\mathcal{N}}, j \in \mathcal{M}, \\
&\qquad y_0^j = z^j \text{ for all } j \in \mathcal{M}.
\end{aligned} \tag{29}$$

By Sion's minimax theorem, we can exchange the maximization and minimization operators so that (29) is equivalent to:

$$\begin{aligned}\max_{y,z}\min_{\theta\geq 0}\ & \sum_{j=1}^{m}\sum_{i=1}^{n}\theta_i^j(p_j^s v_i^j y_i^j-\alpha_i^j(s))\geq 0,\\ \text{s.t. } & \sum_{i=0}^{n} v_i^j y_i^j+z^j=1,\ \text{for all } j\in\mathcal{M},\\ & \sum_{i=0}^{n} y_i^j\leq Kz^j,\ \text{for all } j\in\mathcal{M},\\ & 0\leq y_i^j\leq z^j,\ \text{for all } i\in\bar{\mathcal{N}}, j\in\mathcal{M},\\ & y_0^j=z^j,\ \text{for all } j\in\mathcal{M}.\end{aligned}\tag{30}$$

Therefore, (30) holds if and only if there exist $(y_i^j: i\in\bar{\mathcal{N}}, j\in\mathcal{M})$ and $(z^j: j\in\mathcal{M})$ such that all the constraints in (30) hold and $\sum_{j=1}^m\sum_{i=0}^n\theta_i^j(p_j^s v_i^j y_i^j-\alpha_i^j(s))\geq 0,\quad \forall\theta_i^j\geq 0, i\in\bar{\mathcal{N}}, j\in\mathcal{M}$ holds, which is equivalent to $p_j^s v_i^j y_i^j-\alpha_i^j(s)\geq 0,\quad \forall i,j$. This completes the proof of Proposition 1. □

Proof of Proposition 2

We first show that if (17) holds, $\boldsymbol{\alpha}\in\mathcal{A}(s)$. By the first inequality of (17), we have $v_i^j y_i^j\geq\frac{\alpha_i^j(s)}{p_j^s}$ for all i and j. Plugging this into the first and second equalities of (17), we have

$$1-z^j-z^j v_0^j=\sum_{i=1}^n v_i^j y_i^j\geq\sum_{i=1}^n\frac{\alpha_i^j(s)}{p_j^s}.$$

Thus, by the first and third inequalities of (17), we have

$$\sum_{i'=1}^n\frac{\alpha_{i'}^j(s)}{p_j^s}\leq 1-z^j-z^j v_0^j\leq 1-(1+v_0^j)y_i^j\leq 1-\frac{(1+v_0^j)\alpha_i^j(s)}{p_j^s v_i^j}\ \text{for any } i,j.$$

Rearranging the terms, we have

$$p_j^s\geq\sum_{i'=1}^n\alpha_{i'}^j(s)+\frac{(1+v_0^j)\alpha_i^j(s)}{v_i^j}\ \text{for any } i,j.$$

The first three inequalities of (17) imply that

$$\sum_{i=1}^n\frac{\alpha_i^j(s)}{p_j^s v_i^j}\leq\sum_{i=1}^n y_i^j\leq(K-1)z^j=(K-1)\frac{1-\sum_{i=1}^n v_i^j y_i^j}{1+v_0^j}\leq\frac{K-1}{1+v_0^j}\left(1-\sum_{i=1}^n\frac{\alpha_i^j(s)}{p_j^s}\right),\ \text{for all } j.$$

Rearranging the terms, we have

$$p_j^s\geq\sum_{i=1}^n\alpha_i^j(s)+\frac{1+v_0^j}{K-1}\sum_{i=1}^n\frac{\alpha_i^j(s)}{v_i^j}.$$

Therefore, if (17) holds, we have $\alpha\in\mathcal{A}$.

Next, we show that if $\boldsymbol{\alpha}\in\mathcal{A}(s)$, (17) holds. Given $\alpha\in\mathcal{A}(s)$, define

$$y_i^j=\frac{\alpha_i^j(s)}{p_j^s v_i^j}\ \text{for each } i\in\mathcal{N} \text{ and } j\in\mathcal{M},\ \text{and } y_0^j=z^j=\frac{1}{1+v_0^j}\left(1-\sum_{i=1}^n\frac{\alpha_i^j(s)}{p_j^s}\right).$$

To show (17), it suffices to show the first, second and third inequalities hold because the rest of the constraints are clearly satisfied.

Since $p_j \geq \sum_{i=1}^n \alpha_i^j + \frac{1+v_0^j}{K-1}\sum_{i=1}^n \frac{\alpha_i^j(s)}{v_i^j}$ for all j, we have

$$\sum_{i=1}^n y_i^j = \sum_{i=1}^n \frac{\alpha_i^j(s)}{p_j^s v_i^j} = \frac{1}{p_j^s}\sum_{i=1}^n \frac{\alpha_i^j(s)}{v_i^j} \leq \frac{K-1}{1+v_0^j}\left(1-\sum_{i=1}^n \frac{\alpha_i^j(s)}{p_j^s}\right) = (K-1)z^j = Kz^j - y_0^j \text{ for all } j.$$

Hence, the second inequality of (17) holds. Since $p_j \geq \sum_{i'=1}^n \alpha_{i'}^j + \frac{(1+v_0^j)\alpha_i^j}{v_i^j}$ for any i,j, we have

$$y_i^j = \frac{\alpha_i^j(s)}{p_j^s v_i^j} \leq \frac{1}{1+v_0^j}\left(1-\sum_{i'=1}^n \frac{\alpha_{i'}^j(s)}{p_j^s}\right) = z^j \text{ for all } i\in\mathcal{N}, j\in\mathcal{M}.$$

Therefore, (17) holds. Hence, the feasible region of α in period s is characterized by (18). This completes the proof. □

Proof of Theorem 2

First, by the definition of DWO algorithm for period s, we have that

$$t\alpha_i^j(s) - \sum_{\tau=1}^t y_i^j(S_s^*(\tau)) = d_i^j(t+1,s) \leq \left(d_i^j(t+1,s)\right)^+.$$

Therefore, it suffices to show that, if (11) holds, $\limsup_{t\to+\infty}\frac{1}{t}\cdot\left(d_i^j(t+1,s)\right)^+ \leq 0$ with probability 1.

For $t\geq 1$, define

$$d(t+1,s) := (d_i^j(t+1,s) : i\in\bar{\mathcal{N}}, j\in\mathcal{M}) \text{ and } y(t,s) := (y_i^j(S_s^*(t)) : i\in\bar{\mathcal{N}}, j\in\mathcal{M}).$$

For a vector $x\in\mathbb{R}^n$, we use x^+ to denote the component-wise positive part of x. Note that, for any $A,B\in\mathbb{R}$, $((A+B)^+)^2 \leq (A^+ + B)^2$, we have

$$\begin{aligned}\mathbb{E}||\left(d(t+1,s)\right)^+||_2^2 =& \mathbb{E}||\left(d(t,s)+\alpha(s)-y(t,s)\right)^+||_2^2 \leq \mathbb{E}||(d(t,s))^+ + \alpha(s) - y(t,s)||_2^2 \\ =& \mathbb{E}||(d(t,s))^+||_2^2 + \mathbb{E}||\alpha(s)-y(t,s)||_2^2 + 2\mathbb{E}\left[\sum_{i,j}(d_i^j(t,s))^+\cdot\alpha_i^j(s) - \sum_{i,j}(d_i^j(t,s))^+\cdot y_i^j(S_s^*(t))\right],\end{aligned}$$

where $||\cdot||_2$ denotes the ℓ_2-norm. By (11), since $(d_i^j(t,s))^+\geq 0$ for all i and j,

$$\mathbb{E}[\sum_{i,j}(d_i^j(t,s))^+\cdot\alpha_i^j(s) - \sum_{i,j}(d_i^j(t,s))^+\cdot y_i^j(S_s^*(t))] \leq 0.$$

Furthermore,

$$\mathbb{E}||\alpha(s)-y(t,s)||_2^2 \leq n\cdot m\cdot C, \text{ where } C := \max\mathbb{E}[y_i^j(S)] < +\infty.$$

Therefore,

$$\mathbb{E}||(d(t+1,s))^+||_2^2 \leq ||(d(1,s))^+||_2^2 + tnmC \text{ for all } t\geq 1.$$

By Jensen's inequality and that $||\cdot||_2^2$ is convex,

$$||\mathbb{E}[(d(t+1,s))^+]||_2^2 \leq \mathbb{E}||(d(t+1,s))^+||_2^2 \leq t\cdot(nmC) \text{ for all } t\geq 1.$$

Therefore,

$$\frac{1}{t}||\mathbb{E}[(d(t+1,s))^+]||_2 \leq \sqrt{\frac{1}{t}\cdot(nmC)}, \text{ which implies that } \limsup_{t\to+\infty}\frac{1}{t}||\mathbb{E}[(d(t+1,s))^+]||_2 = 0.$$

Hence

$$\limsup_{t\to+\infty}\frac{1}{t}\cdot(d_i^j(t+1,s))^+ = 0 \text{ with probability 1, for all } i\in\bar{\mathcal{N}} \text{ and } j\in\mathcal{M}.$$

This completes the proof. □

Proof of Theorem 3

Because the algorithm uniformly randomly picks up a time index t,

$$\mathbb{E}[y_i^j(S_s^j(\tilde{G}_{rDWO}(s),\xi_s))] = \frac{1}{T_s}\sum_{\tau=2}^{T_s+1}\mathbb{E}[y_i^j(S_s^{j*}(\tau))].$$

Therefore,

$$\mathbb{E}[y_i^j(S_s^j(\tilde{G}_{rDWO}(s),\xi_s))] - \alpha_i^j(s) = \frac{1}{T_s}\sum_{\tau=2}^{T_s+1}(\mathbb{E}[y_i^j(S_s^{j*}(\tau))] - \alpha_i^j(s)) = -\frac{1}{T_s}\mathbb{E}[d_i^j(T_s+1,s)].$$

By the proof of Theorem 2,

$$\limsup_{T_s\to+\infty}\frac{1}{T_s}\mathbb{E}[d_i^j(T_s+1)] \le \limsup_{T_s\to+\infty}\frac{1}{T_s}\mathbb{E}[d_i^j(T_s+1)]^+ = 0 \text{ for all } i \text{ and } j.$$

Hence, $\mathbb{E}[y_i^j(S_s^j(\tilde{G}_{rDWO}(s),\xi_s))] \ge \alpha_i^j(s)$ for all i and j as $T_s\to+\infty$. This completes the proof. □

Proof of Lemma 1

Let us consider a problem identical to $\mathcal{Q}(\gamma)$ but without budget constraints (i.e., $B_i(\gamma)=+\infty$ for all i and γ), which we denote as $\mathcal{Q}_*(\gamma)$. Consider one version of Algorithm 3 applied to $\mathcal{Q}_*(\gamma)$ with $\alpha_i^j(s)=\alpha_i^{j*}(s)$, where $\boldsymbol{\alpha}^*$ is the solution to (24). In the proof of Lemma 1, let $\tilde{y}_i^j(t,s)\in\mathbb{Z}^+$ (resp. $y_i^j(t,s)\in\mathbb{Z}^+$) denote the number of click-throughs from a type j customer to ad i in period s by user t under the problem $\mathcal{Q}(\gamma)$ (resp. $\mathcal{Q}_*(\gamma)$). It is clear from the construction that the click-through process of $\mathcal{Q}_*(\gamma)$ is identical to that of $\mathcal{Q}(\gamma)$ before stock-out occurs at $\mathcal{Q}(\gamma)$, i.e. $y_i^j(t,s)=\tilde{y}_i^j(t,s)$ for all t before budget stock-out occurs at $\mathcal{Q}(\gamma)$ (i.e., when $b_i\sum_{\tau\le t}\sum_{j\in\mathcal{L}_i}y_i^j(\tau,s) < B_i^*(s,\gamma)$). One should also note that, by construction, at least ad i in period s at least has a budget of

$$B_i^*(s,\gamma) = \frac{\gamma B_i\zeta_s\sum_{j\in\mathcal{L}_i}\alpha_i^{j*}(s)}{\sum_{\sigma=\sigma_i}^{\sigma_i+K_i-1}\zeta_\sigma\sum_{j\in\mathcal{L}_i}\alpha_i^{j*}(\sigma)} = \gamma B_i^*(s).$$

Without loss of generality, we assume $\gamma\in\mathbb{Z}^+$.

We now show that for the system of $\mathcal{Q}_*(\gamma)$,

$$\limsup_{\gamma\uparrow+\infty}\frac{\sum_j\sum_{t=1}^{T_s(\gamma)}y_i^j(t,s)}{T_s(\gamma)} \le \frac{B_i^*(\gamma,s)}{b_iT_s(\gamma)} = \frac{B_i^*(s)}{b_iT_s} \text{ for all } i \text{ and } s.$$

Assume, to the contrary, that,

$$\limsup_{\gamma\uparrow+\infty}\frac{\sum_j\sum_{t=1}^{T_{s_0}(\gamma)}y_{i_0}^j(t,s_0)}{T_{s_0}(\gamma)} > \frac{B_{i_0}^*(s_0)}{b_{i_0}T_{s_0}} \ge \sum_j\alpha_{i_0}^{j*} \text{ for some } i_0 \text{ and } s_0, \tag{31}$$

where the last inequality follows from that $\boldsymbol{\alpha}^*$ solves (14) and the pigeonhole principle. Therefore, by the pigeonhole principle, there exists a j_0 such that $\limsup_{\gamma\uparrow+\infty}\frac{\sum_{t=1}^{T_{s_0}(\gamma)}y_{i_0}^{j_0}(t,s_0)}{T_s(\gamma)} > \alpha_{i_0}^{j_0*}$, i.e., there exists some $\Delta>0$, such that

$$\frac{\sum_{t=1}^{T_{s_0}(\gamma)}y_{i_0}^{j_0}(t,s_0)}{T_{s_0}(\gamma)} > \alpha_{i_0}^{j_0*}+\Delta \text{ for infinitely many } \gamma. \tag{32}$$

Denote the set of γ's that satisfy (32) as Γ. Note that $\frac{1}{T_{s_0}(\gamma)}(\sum_{t=1}^{\tau}y_{i_0}^{j_0}(t,s_0))$ increases by at most $1/(T_{s_0}\gamma)$ as τ increases by 1. Hence, for all $\gamma\in\Gamma$ and $\gamma>3/(T_{s_0}\Delta)$, $\frac{1}{T_{s_0}(\gamma)}(\sum_{t=1}^{\tau}y_{i_0}^{j_0}(t,s_0))$ increases by no more than $\Delta/3$ if τ increases by 1. Therefore, for all $\gamma\in\Gamma$ and $\gamma>3/(T_{s_0}\Delta)$, there exists a $\tau(\gamma)<T_{s_0}(\gamma)$, such that

$$\alpha_{i_0}^{j_0*}(s_0)+\frac{\Delta}{3} < \frac{\sum_{t=1}^{\tau(\gamma)}y_{i_0}^{j_0}(t,s_0)}{T_{s_0}(\gamma)} < \alpha_{i_0}^{j_0*}(s_0)+\frac{2\Delta}{3} \tag{33}$$

By (33), we have that, for infinitely many γ,

$$\sum_{t=1}^{\tau(\gamma)} y_{i_0}^{j_0}(t,s_0) > T_{s_0}(\gamma)\left(\alpha_{i_0}^{j_0*}(s_0)+\frac{\Delta}{3}\right).$$

Hence, for infinitely many γ,

$$(d_{i_0}^{j_0}(t,s_0))^+ = \left((t-1)\alpha_{i_0}^{j_0*}(s_0) - \sum_{\tau=1}^{t-1} y_{i_0}^{j_0}(\tau,s_0)\right)^+ = 0 \text{ for all } t\geq \tau(\gamma)+1,$$

where the equality follows from

$$\sum_{\tau=1}^{t-1} y_{i_0}^{j_0}(\tau,s_0) \geq \sum_{\tau=1}^{\tau(\gamma)} y_{i_0}^{j_0}(\tau,s_0) > T_{s_0}(\gamma)\alpha_{i_0}^{j_0*}(s_0) > (t-1)\alpha_{i_0}^{j_0*}(s_0).$$

Therefore, ad i_0 will not be offered to customer type j_0 in period s_0 for all $t\geq\tau(\gamma)+1$. Hence, $y_{i_0}^{j_0}(t,s_0)=0$ for all $t\geq\tau(\gamma)+1$ and $t\leq T_{s_0}(\gamma)$. By (33), we have

$$\frac{\sum_{t=1}^{T_s(\gamma)} y_{i_0}^{j_0}(t,s_0)}{T_{s_0}(\gamma)} = \frac{\sum_{t=1}^{\tau(\gamma)} y_{i_0}^{j_0}(t,s_0)}{T_{s_0}(\gamma)} < \alpha_{i_0}^{j_0*}(s_0)+\frac{2\Delta}{3} \text{ for } \gamma\in\Gamma \text{ and } \gamma>3/(T_{s_0}\Delta),$$

which contradicts inequality (32). Therefore, for the system of $\mathcal{Q}_*(\gamma)$, we have

$$\limsup_{\gamma\uparrow+\infty} \frac{\sum_j\sum_{t=1}^{T_s(\gamma)} y_i^j(t,s)}{T_s(\gamma)} \leq \frac{B_i^*(s,\gamma)}{b_iT_s(\gamma)} = \frac{B_i^*(s)}{b_iT_s} \text{ for all } i \text{ and } s. \tag{34}$$

On the other hand, Theorem 2 then implies that, for the system of $\mathcal{Q}_*(\gamma)$, we have

$$\liminf_{\gamma\to+\infty} \frac{1}{T_s(\gamma)} \sum_{t=1}^{T_s(\gamma)} y_i^j(t,s) \geq \alpha_i^{j*}(s) \text{ for each } i\in\bar{\mathcal{N}},\ j\in\mathcal{M}, \text{ and } 1\leq s\leq\Sigma. \tag{35}$$

Furthermore, since the click-through process of $\mathcal{Q}(\gamma)$ is identical to that of $\mathcal{Q}_*(\gamma)$ before budget stock-out occurs at $\mathcal{Q}(\gamma)$. Therefore, for the system of $\mathcal{Q}(\gamma)$, a standard coupling argument implies that and (34) and (35) hold as well. This concludes the proof of the lemma. $\square$

Before proving Theorem 4, we propose the CBLP policy in our ad allocation context. We define the following linear program for problem $\mathcal{Q}(\gamma)$:

$$\begin{aligned}
\max\ & \sum_{s,i,j,S\in\mathcal{S}_j} T(\gamma)\zeta_s r_i p_j^s \phi_i^j(S) z_s^j(S) \\
s.t.\ & \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s(\gamma) b_i \sum_{j,S} p_j^s\phi_i^j(S)z_s^j(S) \leq B_i(\gamma) \text{ for all } i\in\mathcal{N} \\
& \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s(\gamma) \sum_{j\in c,S\in\mathcal{S}_j} p_j^s\phi_i^j(S)z_s^j(S) \geq \eta_i^c(\gamma) \text{ for all } i\in\bar{\mathcal{N}} \text{ and } c\subset\mathcal{M} \\
& \sum_{S\in\mathcal{S}_j} z_s^j(S) \leq 1 \text{ for all } j\in\mathcal{M} \text{ and } s \\
& z_s^j(S)\geq 0 \text{ for all } j\in\mathcal{M}, S\in\mathcal{S}_j, \text{ and } s
\end{aligned} \tag{$\mathcal{LP}(\gamma)$}$$

We define the optimal objective value of $\mathcal{LP}(\gamma)$ as $Rew_{LP}^*(\gamma)$. Denote the solution to $\mathcal{LP}(\gamma)$ as $\mathbf{z}^*$. The CBLP policy, denoted as $\tilde{G}_{CBLP}$, is to randomly display offer set S with probability $z_s^{j*}(S)$ when a customer of type j arrives in period s. As an extension to Propositions 1 and 2 of Liu and Van Ryzin (2008), we have the following lemma, which implies that the optimal value obtained by CBLP is an upper bound of the optimal reward of the original ad allocation problem, and that this upper bound is asymptotically tight.

LEMMA 2. *For problem $\mathcal{Q}(\gamma)$, $Rew^*(\gamma) \le Rew^*_{LP}(\gamma)$ for all $\gamma > 0$, and*

$$\lim_{\gamma \uparrow +\infty} \frac{1}{\gamma} Rew^*(\gamma) = \lim_{\gamma \uparrow +\infty} \frac{1}{\gamma} Rew^*_{LP}(\gamma) \tag{36}$$

Proof of Lemma 2

We first show that $Rew^*(\gamma) \le Rew^*_{LP}(\gamma)$. We remark that the argument is the same as the proof of Proposition 1 in Liu and Van Ryzin (2008). Define the policy that solves the stochastic program (1) in the paper as $\tilde{G}^*$ that achieves $Rew^*(\gamma)$. We first check the following two claims:

1. $\tilde{G}^*$ could induce a feasible solution to the LP $\mathcal{LP}(\gamma)$.
2. The solution induced by $\tilde{G}^*$ to $\mathcal{LP}(\gamma)$ gives rise to the objective function value of $Rew^*(\gamma)$.

The above two claims imply that $Rew^*(\gamma)$ must be bounded from above by the optimal objective function value of $\mathcal{LP}(\gamma)$, i.e., $Rew^*(\gamma) \le Rew^*_{LP}(\gamma)$.

We now prove Claim 1. Let $\{S^*_s(t) : 1 \le s \le \Sigma, 1 \le t \le T_s\}$ be the sequence of (randomized) offer set, under the optimal policy $\tilde{G}^*$, displayed to customer t in period s. Based on $\tilde{G}^*$, we construct a solution to $\mathcal{LP}(\gamma)$:

$$z^j_s(S) := \mathbb{E}[\mathbf{1}\{S^*_s(t) = S\} | \xi^t_s = j] \text{ for all } s,\ j, \text{ and } S. \tag{37}$$

Because $\tilde{G}^*$ is feasible for the stochastic program (1), we take the expectation of all sample-path constraints in (1) to obtain that $z^j_s(S)$ given by (37) satisfy all the constraints of $\mathcal{LP}(\gamma)$ hold. Therefore, $\tilde{G}^*$ induces a feasible solution to $\mathcal{LP}(\gamma)$.

Next, since $\tilde{G}^*$ is optimal for (1), we have

$$Rew_*(\gamma) = \mathbb{E}\left[\sum_{s=1}^{\Sigma} \sum_{t=1}^{T_s} \sum_{i \in \tilde{\mathcal{N}}_s} r_i y_i^{\xi^t_s} | \tilde{G}^* \right]$$

Therefore, $(z^j_s(S) : j \in \mathcal{M}, S \in \mathcal{S}_j, 1 \le s \le \Sigma)$ induced by (37) gives rise to the objective function value of $Rew^*(\gamma)$ for the LP $\mathcal{LP}(\gamma)$. This proves Claim 2 and, thus, $Rew^*(\gamma) \le Rew^*_{LP}(\gamma)$.

Next, we prove asymptotic optimality (36) holds. We remark that the argument is similar to the proof of Proposition 2 in Liu and Van Ryzin (2008).

Let $(z^{j*}_s(S) : j \in \mathcal{M}, S \in \mathcal{S}_j, 1 \le s \le \Sigma)$ be an optimal solution to $\mathcal{LP}(\gamma)$, which is independent of the scale parameter γ. Hence, $\frac{1}{\gamma} Rew^*_{LP}(\gamma)$ is a constant independent of γ. For each scaling parameter γ, define the CBLP policy $\tilde{G}_*(\gamma)$ under which in each period s, when a type-j customer arrives, the platform displays offer set S with probability $z^{j*}_s(S)$.

Next, we show that $\tilde{G}_*(\gamma)$ is feasible for the scaled stochastic program (2) as $\gamma \uparrow +\infty$. More specifically, under the policy $\tilde{G}_*(\gamma)$ and $\gamma \uparrow +\infty$,

$$\begin{aligned} \frac{1}{T(i,\gamma)} \sum_{s=\sigma_i}^{\sigma_i+K_i-1} \sum_{t=1}^{T_s(\gamma)} b_i y_i^{\xi^t_s}(S^t_s) =& \frac{1}{T(i,\gamma)} \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s(\gamma) \cdot \frac{1}{T_s(\gamma)} \sum_{t=1}^{T_s(\gamma)} b_i y_i^{\xi^t_s}(S^t_s) \\ &\to \frac{1}{T(i,\gamma)} b_i \sum_{s=\sigma_i}^{\sigma_i+K_i-1} T_s(\gamma) \sum_{j,S} p^s_j \phi^j_i(S) z^{j*}_i(S) \\ &\le \frac{B_i(\gamma)}{T(i,\gamma)}, \end{aligned} \tag{38}$$

where $\to$ follows from the law of large numbers and the last inequality follows from $(z_i^{j*}(S): j \in \mathcal{M}, S \in \mathcal{S}_j, 1 \le s \le \Sigma)$ is feasible for $\mathcal{LP}(\gamma)$. Similarly, we have, under the policy $\tilde{G}_*(\gamma)$ and $\gamma \uparrow +\infty$,

$$\frac{1}{T(i,\gamma)} \sum_{s=\sigma_i}^{\sigma_i+K_i-1} \sum_{t=1}^{T_s(\gamma)} \sum_{j \in c} y_i^j(S_s^t) \mathbf{1}\{\xi_s^t = j\} \ge \frac{\eta_i^c(\gamma)}{T(i,\gamma)} \tag{39}$$

Inequalities (38) and (39) together imply that the CBLP policy $\tilde{G}_*(\gamma)$ is feasible as $\gamma \uparrow +\infty$.

Finally, we show that

$$\lim_{\gamma \to +\infty} \frac{1}{\gamma} Rew(\gamma, \tilde{G}_*(\gamma)) = Rew_{LP}^*(1).$$

Under the CBLP policy $\tilde{G}_*(\gamma)$,

$$\begin{aligned} \frac{1}{\gamma} Rew(\gamma, \tilde{G}_*(\gamma)) =& \frac{1}{\gamma} \mathbb{E}\left[\sum_{s=1}^{\Sigma} \sum_{t=1}^{T_s(\gamma)} \sum_{i \in \bar{\mathcal{N}}_s} r_i y_i^{\xi_s^t}(S_s^t) \right] \\ =& \frac{1}{\gamma} \sum_{s=1}^{\Sigma} \sum_{t=1}^{T_s(\gamma)} \sum_{i \in \bar{\mathcal{N}}_s} r_i p_j^s \phi_i^j(S) z_s^{j*}(S) \\ =& \frac{1}{\gamma} \sum_{s,i,j,S \in \mathcal{S}_j} T\gamma \zeta_s r_i p_j^s \phi_i^j(S) z_s^{j*}(S) \\ =& Rew_{LP}^*(1). \end{aligned} \tag{40}$$

Furthermore, we have

$$\frac{1}{\gamma} Rew(\gamma, \tilde{G}_*(\gamma)) = Rew_{LP}^*(1) = \frac{1}{\gamma} Rew_{LP}^*(\gamma) \ge \frac{1}{\gamma} Rew^*(\gamma),$$

where the inequality follows from the first part of this lemma. By definition,

$$\lim_{\gamma \to +\infty} \frac{1}{\gamma} Rew(\gamma, \tilde{G}_*(\gamma)) \le \lim_{\gamma \to \infty} \frac{1}{\gamma} Rew^*(\gamma),$$

so (36) holds. This concludes the proof of Lemma 2. □

Proof of Theorem 4

Because $Rew(\gamma, \tilde{G}_{DWO-B}) \le Rew^*(\gamma)$, it suffices to show that

$$\liminf_{\gamma \to +\infty} \frac{Rew(\gamma, \tilde{G}_{DWO-B})}{Rew^*(\gamma)} = 1.$$

First, note that as $\gamma \uparrow +\infty$, by Lemma 1,

$$\liminf_{\gamma \to +\infty} \frac{1}{T_s(\gamma)} \sum_{t=1}^{T_s(\gamma)} y_i^j(t,s) \ge \alpha_i^{j*}(s),$$

we must have

$$\liminf_{\gamma \to +\infty} \frac{Rew^*(\gamma)}{T(\gamma)} \ge \liminf_{\gamma \to +\infty} \frac{Rew(\gamma, \tilde{G}_{DWO-B})}{T(\gamma)} \ge \sum_{s,i,j} \zeta_s r_i \alpha_i^{j*}(s), \tag{41}$$

where the last inequality follows from Lemma 1 and the fact that that, in each period s, the budget for ad i is at least as much as $B_i^*(s,\gamma)$. Hence, it suffices to show that

$$\liminf_{\gamma \to +\infty} \frac{T(\gamma) \sum_{s,i,j} \zeta_s r_i \alpha_i^{j*}(s)}{Rew^*(\gamma)} = 1.$$

By Lemma 2, $Rew^*_{LP}(\gamma) \geq Rew^*(\gamma)$. Denote the optimal solution to $\mathcal{LP}(\gamma)$ as $z^*(S) := (z_s^{j*}(S) : j \in \mathcal{M}, S \in \mathcal{S}_j, 1 \leq s \leq \Sigma)$. Clearly, $z^*(S)$ corresponds to a randomized CBLP offer set policy that generates click-through goals $\tilde{\alpha}_i^{j*}(s) = \sum_S p_j^s \phi_i^j(S) z_s^{j*}(S)$. It is straightforward to check that $(\tilde{\alpha}_i^{j*}(s) : i \in \bar{\mathcal{N}}_s, j \in \mathcal{M}, 1 \leq s \leq \Sigma)$ is a feasible solution to (14). Thus,

$$\sum_{s,i,j} \zeta_s r_i \alpha_i^{j*}(s) \geq \sum_{s,i,j} \zeta_s r_i \tilde{\alpha}_i^{j*}(s) = \lim_{\gamma \to +\infty} \frac{1}{T(\gamma)} \cdot Rew^*(\gamma), \tag{42}$$

where the last equality follows from Lemma 2. It follows immediately from combining inequalities (41) and (42) that

$$\liminf_{\gamma \to +\infty} \frac{T(\gamma) \sum_{s,i,j} \zeta_s r_i \alpha_i^{j*}(s)}{Rew^*(\gamma)} = 1.$$

This completes the proof of Theorem 4. □

Proof of Proposition 3

First, we need to show that Algorithm $\tilde{G}_{DWO-ART}$ satisfies that

$$\liminf_{\gamma \to +\infty} \frac{1}{T_s(\gamma)} \sum_{t=1}^{T_s(\gamma)} \sum_{j \in \mathcal{M}} r_i y_i^j (S_s^*(t, \tilde{G}_{DWO-ART})) \geq \sum_{j \in \mathcal{M}} r_i \alpha_i^{j*}(s) \text{ for all } i \in \bar{\mathcal{N}} \text{ and } 1 \leq s \leq \Sigma. \tag{43}$$

Note that

$$\begin{aligned}
&\mathbb{E}||\big(d(t+1,s)\big)^+||_2^2 \\
=&\mathbb{E}||\big(d(t,s) + diag(r^T \alpha^*(s)) - diag(r^T \tilde{y}(t,s))\big)^+||_2^2 \leq \mathbb{E}||(d^2(t,s))^+ + diag(r^T \alpha^*(s)) - diag(r^T \tilde{y}(t,s))||_2^2 \\
=&\mathbb{E}||(d(t,s))^+||_2^2 + \mathbb{E}||diag(r^T \alpha^*(s)) - diag(r^T y(t,s))||_2^2 + 2\mathbb{E}[\sum_{i,j} (d_i(t,s))^+ \cdot r_i^j \alpha_i^{j*}(s) - \sum_{i,j} (d_i(t,s))^+ \cdot r_i^j y_i^j (S_s^*(t))],
\end{aligned}$$

where r denotes the matrix of $(r_i^j = r_i : i \in \bar{\mathcal{N}}, j \in \mathcal{M})$ and $diag(\cdot)$ denotes the diagonal element vector of a matrix. Since $\alpha^*(s)$ satisfies $h_s(\theta|\boldsymbol{\alpha}(s)) \geq 0$ for all $\theta \geq 0$, we have

$$\mathbb{E}[\sum_{i,j} (d_i(t,s))^+ \cdot r_i^j \alpha_i^{j*}(s) - \sum_{i,j} (d_i(t,s))^+ \cdot r_i^j y_i^j (S_s^*(t, \tilde{G}_{DWO-ART}))] \leq 0.$$

Furthermore, we have the following bound

$$\mathbb{E}||diag(r^T \cdot \alpha^*(s)) - diag(r^T \cdot \tilde{y}(t,s))||_2^2 \leq (\max_{i,j}\{r_i^j\})^2 \cdot (nm).$$

Therefore, by a similar argument to the proof of Theorem 2, we have

$$\frac{1}{t} ||\mathbb{E}[(d(t+1,s))^+]||_2 \leq \max_{i,j}\{r_i^j\} \sqrt{\frac{1}{t} \cdot (nmC)},$$

where $C := \max \mathbb{E}[y_i^j(S)]$, which implies that

$$\limsup_{t \to +\infty} \frac{1}{t} \cdot (d_i(t+1,s))^+ = 0 \text{ with probability 1, for all } i \in \mathcal{N}.$$

In other words, we have, for all $i \in \mathcal{N}$,

$$\liminf_{\gamma \to +\infty} \frac{1}{T_s(\gamma)} \sum_{t=1}^{T_s(\gamma)} \sum_{j \in \mathcal{M}} r_i y_i^j (S_s^*(t, \tilde{G}_{DWO-ART})) - \sum_j r_i \alpha_i^{j*}(s) = -r_i \limsup_{t \to +\infty} \frac{1}{t} d_i(t+1,s) \geq -\limsup_{t \to +\infty} \frac{1}{t} (d_i(t+1,s))^+ = 0,$$

which proves inequality (43).

We are now ready to show that

$$\lim_{\gamma\to+\infty}\frac{Rew(\gamma,\tilde{G}_{DWO-ART})}{Rew^*(\gamma)}=1.$$

By definition, we have $Rew(\gamma,\tilde{G}_{DWO-ART})\le Rew^*(\gamma)$. Thus, it suffices to show that

$$\liminf_{\gamma\to+\infty}\frac{Rew(\gamma,\tilde{G}_{DWO-ART})}{Rev^*(\gamma)}=1.$$

First, note that as $\gamma\uparrow+\infty$, by (43),

$$\liminf_{\gamma\to+\infty}\frac{1}{T(\gamma)}\sum_{s=1}^{\Sigma}\sum_{t=1}^{T_s(\gamma)}\sum_{j\in\mathcal{M}}r_i y_i^j(S_s^*(t,\tilde{G}_{DWO-ART}))\ge\sum_s\zeta_s\sum_j r_i\alpha_i^{j*}(s),\ \text{for all } i\in\bar{\mathcal{N}}.$$

We must have

$$\liminf_{\gamma\to+\infty}\frac{Rew^*(\gamma)}{T(\gamma)}\ge\liminf_{\gamma\to+\infty}\frac{Rew(\gamma,\tilde{G}_{DWO-ART})}{T(\gamma)}=\liminf_{\gamma\to+\infty}\frac{\sum_s\sum_{t=1}^{T_s(\gamma)}\sum_{s,i,j}y_i^j(S_s^*(t,\tilde{G}_{DWO-ART})}{T(\gamma)}\ge\sum_s\sum_i(\sum_j\zeta_s r_i\alpha_i^{j*}(s)). \tag{44}$$

As we have shown in the Proof of Theorem 4,

$$\liminf_{\gamma\to+\infty}\frac{T(\gamma)\sum_{s,i,j}\zeta_s r_i\alpha_i^{j*}(s)}{Rew^*(\gamma)}=1.$$

Together with (44), we have

$$\liminf_{\gamma\to+\infty}\frac{Rew(\gamma,\tilde{G}_{DWO-ART})}{Rew^*(\gamma)}\ge\liminf_{\gamma\to+\infty}\frac{T(\gamma)\sum_{s,i,j}\zeta_s r_i\alpha_i^{j*}(s)}{Rew^*(\gamma)}=1.$$

This completes the proof of Proposition 3. □

Appendix D. Other Debt-Weighted Offer-set Policies

In this section, we give a few other debt-weighted offer-set policies. To begin with, we propose the following algorithm as an analog of $\tilde{G}_{DWO}$ for non-personalized click-through goals.

Algorithm 5 NON-PERSONALIZED DEBT-WEIGHTED OFFER-SET (NDWO) POLICY

Initialize: Period s and $d_i(1,s)\leftarrow 0$ for all ad $i\in\bar{\mathcal{N}}$.

For each customer $t\ge 1$**:**

1: Observe the customer type $\xi_s^t=\hat{j}$.

2: Display offer set $S_s^*(t)$ to the customer, where

$$S_s^*(t)\leftarrow\arg\max_S\sum_{i\in S}(d_i(t,s))^+\,\phi_i^{\hat{j}}(S) \tag{45}$$

3: Observe customer click-throughs $(y_i^j(t):i\in\mathcal{N},j\in\mathcal{M})$.

4: $d_i(t+1,s)\leftarrow d_i(t,s)+\alpha_i(s)-\sum_{j\in\mathcal{M}}y_i^j(t,s)$ for all $i\in\bar{\mathcal{N}}$.

We consider the following constraints of ad-specific click-through goals of period s:

$$\mathbb{E}[\sum_{j\in\mathcal{M}}y_i^j(S_s^j(\tilde{G}_s,\xi_s))]\ge\alpha_i(s),\ \text{for each } i\in\bar{\mathcal{N}}, \tag{46}$$

where $\alpha_i(s) \ge 0$ is the (total) click-through goal for ad i in period s. The following proposition characterizes the necessary and sufficient condition for feasible non-personalized click-through goals (in the second stage) and demonstrate the validity of Algorithm 5 to achieve the goals.

PROPOSITION 4. *Given the non-personalized click-through goals* $(\alpha_i(s) \ge 0 : i \in \bar{\mathcal{N}})$, *we have:*

(a) The click-through constraints (46) *are feasible if and only if:*

$$\max_{G_s \in \mathcal{G}_s} \sum_{i,j} p_j^s \phi_i^j(S_s^j(G_s, j))\theta_i \ge \sum_i \alpha_i(s)\theta_i, \text{ for all } \theta_i \ge 0 \; i \in \bar{\mathcal{N}} \tag{47}$$

(b) If (47) *holds, Algorithm 5 satisfies the non-personalized click-through goals* $(\alpha_i : i \in \bar{\mathcal{N}})$.

Next, we propose the following algorithm which prescribes the dynamic offer set policy with individualized reward goals. Following the same proof as Theorem 3, one can show that the DWO-RT policy (Algorithm 6 below) is asymptotically optimal for problem $\mathcal{Q}(\gamma)$ as the problem size γ increases to infinity.

Algorithm 6 DEBT-WEIGHTED OFFER-SET WITH REWARD TARGET (DWO-RT) POLICY $\tilde{G}_{DWO-RT}$

First-stage optimization: Solve (14) to obtain the optimal sales target $\boldsymbol{\alpha}^*$ and the budget $B_i^*(s)$.

For each period $s \ge 1$**:**

Initialize: $d_i^j(1,s) \leftarrow 0$ for all $i \in \mathcal{N}$, $j \in \mathcal{M}$, and s.

For each customer $t \ge 1$**:**

1: Observe the customer type $\xi_s^t = \hat{j}$.

2: Display offer set $S_s^*(t)$ to the customer, where

$$S_s^*(t) \leftarrow \arg\max_S \sum_{i \in S} \left(d_i^{\hat{j}}(t,s)\right)^+ \phi_i^{\hat{j}}(S). \tag{48}$$

3: Observe the customer click-throughs $(y_i^j(t,s) : i \in \bar{\mathcal{N}}_s, j \in \mathcal{M})$. The reward, $\sum_i r_i y_i^{\hat{j}}(t,s)$, is collected. In the case where the budget for ad i in period s is exhausted, i.e., $(\sum_j \sum_{\tau \le t} y_i^j(\tau,s))b_i = B_i^*(s)$, any offer set containing this ad will no longer be displayed hereafter in period s.

4: $d_i^j(t+1,s) \leftarrow d_i^j(t,s) + r_i\alpha_i^{j*}(s) - r_i y_i^j(t,s)$ for all $i \in \bar{\mathcal{N}}$ and $j \in \mathcal{M}$.

Proof of Proposition 4

Part (a). To obtain the feasibility condition, note that (46) is equivalent to the following linear program:

$$\begin{aligned} &\max_{\mu_s(\cdot)} 0 \\ &\text{s.t.} \sum_{G_s \in \mathcal{G}_s} \mu_s(G_s) \sum_j p_j^s \phi_i^j(S_s^j(G_s,j)) \ge \alpha_i(s), \text{ for each } i \in \bar{\mathcal{N}} \\ &\qquad \sum_{G_s \in \mathcal{G}_s} \mu_s(G_s) = 1 \\ &\qquad \mu_s(G_s) \ge 0 \text{ for all } G_s \in \mathcal{G}_s. \end{aligned} \tag{49}$$

The dual of (49) can be written as:

$$\begin{aligned} &\min_{\theta_0,\theta_i} \{\theta_0 - \sum_{i\in\bar{\mathcal{N}}} \alpha_i(s)\theta_i\} \\ &s.t. \sum_{j\in\mathcal{M}, i\in\bar{\mathcal{N}}} p_j^s \phi_i^j(S_s^j(G_s,j))\theta_i - \theta_0 \le 0, \text{ for all } G_s \in \hat{G}_s; \\ &\quad \theta_i \ge 0 \text{ for all } i \in \bar{\mathcal{N}}. \end{aligned} \tag{50}$$

By strong duality, (49) is feasible if and only if $\min_{\theta_i\ge 0}\left\{\max_{G_s\in\mathcal{G}_s}\sum_{i\in\bar{\mathcal{N}},j\in\mathcal{M}} p_j^s\phi_i^j(S_s^j(G_s,j))\theta_i - \sum_{i\in\bar{\mathcal{N}}}\alpha_i(s)\theta_i\right\} \ge 0$, which is equivalent to (47). This proves proves part (a).

Part (b). Part (b) of Proposition 4 follows from the same argument as the proof of Theorem 2, so we only sketch the proof. Specifically, it suffices to show that if (47) holds, $d_i(t+1)/t \to 0$ in probability for all i, where $d_i(t)$ is defined in Algorithm 5 for each i and t. To show that $d_i(t+1)/t \to 0$ in probability, we leverage the condition (47) and find that $||\mathbb{E}[(d(t+1))^+]||_2^2 \le tnC$, which implies that

$$\limsup_{t\to+\infty} \frac{1}{t}\cdot(d_i(t+1)) = 0 \text{ with probability 1 for all } i \in \bar{\mathcal{N}}.$$

This proves Part (b). □

Appendix E. Advertising in Alibaba

Alibaba Group provides a marketing platform called Alimama (`https://www.alimama.com/index.htm#!/home/index`) to help sellers advertising in Alibaba e-commerce platforms such as Tmall.com. Figure 4 shows the structure of an advertising tool in Alimama. An advertiser can create an ad campaign plan, specifying the keywords, targeted groups of people, and the average bid price per click-through. The advertiser can then upload the creative of the ad (i.e., the ad content).

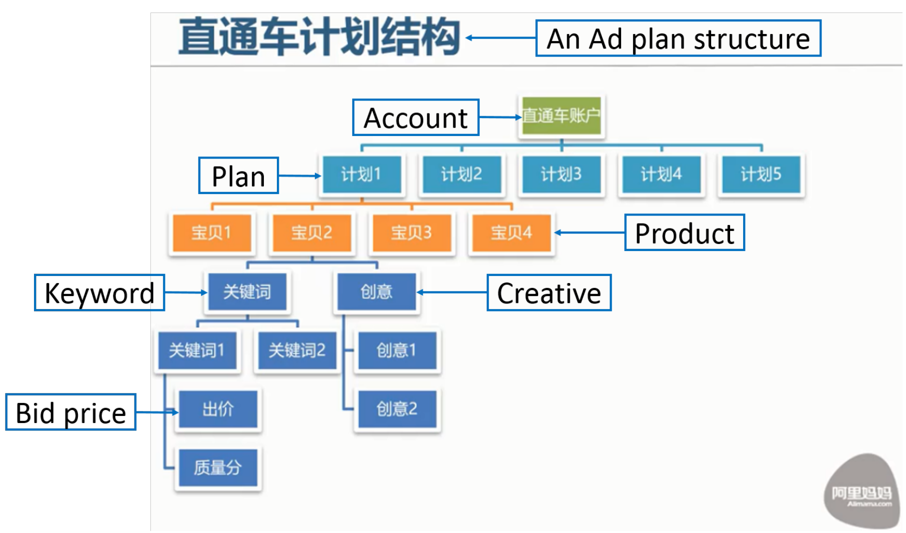
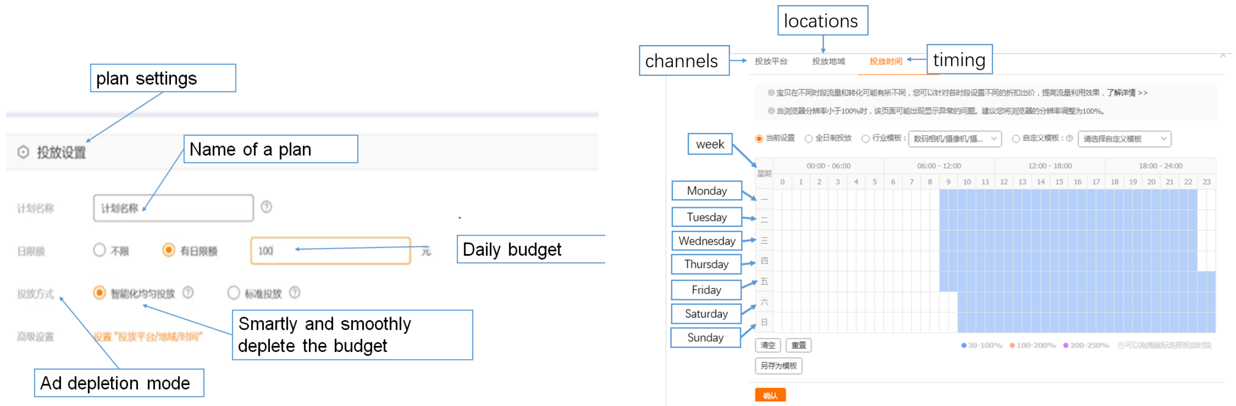


Figure 4 **An Ad plan structure provided by Alimama**

Figures 5(a) and 5(b) show details of a campaign plan settings. As shown in Figure 5(a), an advertiser can set a daily budget of advertising. Alimama will automatically stop allocating user impressions to this



(a) Settings of an Ad plan (b) Time settings

Figure 5 Settings of an Ad campaign

ad in the rest of the day once its daily budget is depleted. It also allows advertisers to set advertising mode, such as "Smartly and smoothly deplete the budget".

Alimama allows advertisers to set the specific time intervals when an ad should be displayed to customers (i,e, the active time intervals for the ad). Figure 5(b) illustrates a screen shot for setting up the active time intervals of an ad in a week. From Monday to Thursday, the advertiser only advertises on Alibaba from 9am to 23pm (the light blue blocks). On Friday, the advertiser keeps its ad active from 9am til mid night. In the weekend, the active time is set from 10am to mid night. The ad is not active during the time intervals in white. In fact, advertisers can set the active time intervals of their ads based on the estimated traffic of the targeted customer groups.